# A Bayesian Approach to Phylogenetic Networks

Radice, Rosalba

*Award date:*
2011

*Awarding institution:*
University of Bath

# A Bayesian Approach to Phylogenetic Networks

submitted by

Rosalba Radice

for the degree of Doctor of Philosophy

of the

University of Bath

Department of Mathematical Sciences

January 2011

# Contents

# List of Figures

# List of Tables

# Acknowledgements

I would like to thank my funders, Bath University Studentship and EPSRC DTA, for their financial support throughout my studies.

# Summary

Traditional phylogenetic inference assumes that the history of a set of taxa can be explained by a tree. This assumption is often violated as some biological entities can exchange genetic material giving rise to non-treelike events often called reticulations. Failure to consider these events might result in incorrectly inferred phylogenies, and further consequences, for example stagnant and less targeted drug development. Phylogenetic networks provide a flexible tool which allow us to model the evolutionary history of a set of organisms in the presence of reticulation events. In recent years, a number of methods addressing phylogenetic network reconstruction and evaluation have been introduced. One of such methods has been proposed by Moret *et al*. (2004). They defined a phylogenetic network as a directed acyclic graph obtained by positing a set of edges between pairs of the branches of an underlying tree to model reticulation events. Recently, two works by Jin *et al*. (2006), and Snir and Tuller (2009), respectively, using this definition of phylogenetic network, have appeared. Both works demonstrate the potential of using maximum likelihood estimation for phylogenetic network reconstruction. We propose a Bayesian approach to the estimation of phylogenetic network parameters. We allow for different phylogenies to be inferred at different parts of our DNA alignment in the presence of reticulation events, at the species level, by using the idea that a phylogenetic network can be naturally decomposed into trees. A Markov chain Monte Carlo algorithm is provided for posterior computation of the phylogenetic network parameters. Also a more general algorithm is proposed which allows the data to dictate how many phylogenies are required to explain the data. This can be achieved by using stochastic search variable selection. Both algorithms are tested on simulated data and also demonstrated on the ribosomal protein gene $rps11$ data from five flowering plants. The proposed approach can be applied to a wide variety of problems which aim at exploring the possibility of reticulation events in the history of a set of taxa.

# Chapter 1

# Introduction

## 1.1 Motivation

Phylogenenies are the main tool for representing evolutionary histories of species. Reconstructing phylogenies is a major component of modern research programs in many areas of biology and medicine. Real-world interest is strong in determining such relationships. For example, pharmaceutical companies may use phylogeny reconstruction in drug discovery for finding plants with similar gene production or use phylogeny reconstruction to develop vaccines, antimicrobials and herbicides. Also, the reconstruction of phylogenies is a crucial tool as it allows the researcher to test new models of evolution.

Biologists, mathematicians, statisticians, and computer scientists have developed a number of methods for reconstructing these relationships, with the usual model being a phylogenetic tree. However, it is widely understood and accepted that the evolutionary history of some species is not really a tree (see Linder *et al.*, 2004 and references therein). Rather it is a network in which there have been a large number of reticulate evolutionary events (i.e. exchange of genetic material between two or more taxa). These are among the fundamental processes creating diversity at the gene level, particularly among bacteria (Heuer and Smalla, 2007) and plants (Bergthorsson *et al.*, 2003, 2004; Linder and Rieseberg, 2004), and causing antibiotic resistance genes in the environment (see Martínez *et al.*, 2007 and references therein) which is a major factor that limits the effectiveness of antibiotics. Failure to detect reticulation might result in incorrectly inferred phylogenies and hence invalidate the conclusions of research studies. As a consequence, appropriate tools for inferring robust phylogenies are required.

In recent years many researchers have introduced a number of tools to ad-

dress phylogenetic network reconstruction and evaluation, leading to a variety of methods. In particular, one of these methodologies for network reconstruction has been proposed by Moret *et al*. (2004). They defined a phylogenetic network as a directed acyclic graph (DAG) obtained by positing a set of edges between pairs of the branches of an underlying tree (species tree) to model reticulation events. Recently, two works by Jin *et al*. (2006), and Snir and Tuller (2009), respectively, using this definition of phylogenetic network, have appeared. Both works demonstrate the potential of using maximum likelihood (ML) estimation for phylogenetic network reconstruction. To date, no equivalent Bayesian method for phylogenetic network estimation has been developed. Here we propose a Bayesian approach to phylogenetic networks which is a promising alternative to the above methods. In fact a Bayesian framework can produce more straightforward statistical measures of phylogeny, can be made computationally faster at least if compared to a ML approach with bootstrap replicates, and careful prior specification can take into account biological information that is otherwise inadmissible with any other approach.

## 1.2 Contribution of the thesis

The **objective** of this thesis is the development of a Bayesian modelling framework for phylogenetic networks. Throughout the thesis we extensively test and investigate the performance of this approach. We show that this framework can recover the true synthetic parameter values, and can be robust to several model specifications. We apply it to the ribosomal protein gene *rps11* of flowering plants (Bergthorsson *et al*., 2003). The findings show that significant variation caused by reticulation events in this phylogeny is detected.

**Another contribution** of the thesis is the development of a more flexible and general algorithm which allows the data to dictate how many phylogenies are required to explain the data. This is achieved by using a sampling scheme based on a variable selection method, originally introduced and developed by George and McCulloch (1993) for linear regression models. The good performance of the proposed method is illustrated on simulations and the *rps11* data.

**We also conduct** a review of the current state of the art in phylogenetic networks which provides an accessible introduction to the lastest developments in this field hoping that it can serve as an impetus to inspire further research and creativity in this fascinating area.

Most of the material of the thesis will serve as the basis for journal articles.

## 1.3 Organization of the thesis

The contents of the present dissertation can be summarised as follows. In Chapter 2 we introduce some notation and definitions which are essential tools in later chapters, and describe some results which are used throughout the thesis. In particular we recall some basic concepts of phylogenetics and related statistical tools such as Markov models and likelihood function of a phylogenetic tree.

In Chapter 3 we begin by providing a background for reticulation events, discussing the consequences of unmodelled reticulation events, and giving some references of works in related problems. Then, we describe the biology underlying reticulation events. In particular we review the concept of horizontal gene transfer (HGT) and hybrid speciation (HS). We also review existing methods to model these events in the literature, specifically the naive ML estimation, ML with hidden Markov models (HMMs), and maximum parsimony (MP). The aim is to provide an accessible introduction to this field of research, trying to achieve a good balance between mathematical tractability and intuition. We conclude by discussing advantages and limitations of these methods.

Chapter 4 presents a Bayesian approach to phylogenetic networks to model reticulation events at the species level. Markov chain Monte Carlo (MCMC) techniques are employed to compute all posterior quantities of the evolution model, and allow inferences to be made regarding the number of different phylogenies for different parts of DNA sequences. In particular, to model different phylogenies at each site two approaches are considered: naive, where the sites are modelled independently and a structure of HMMs, which accounts for the fact that reticulation events generally affect a number of adjacent sites. Also, the stochastic forward-backward algorithm, which is a single component block procedure is contrasted to the Gibbs sampler, which is a large component block procedure.

Chapter 5 investigates the performance of the method described in the previous chapter on simulated data. Specifically we test the algorithm on its ability to recover the true synthetic parameter values, and correctly classify tree topologies along aligned sequences. We also contrast the naive prior for the sequence of topologies to the hidden Markov model. Then we compare the

performance of Gibbs sampling and the forward-backward algorithm for the sequence phylogenies in terms of convergence and mixing. Finally we present several scenarios and misspecifications with the aim of gaining insight in terms of practical implications.

In Chapter 6 we apply the method to real data. We first introduce the concept of lateral gene transfer in plants, explain the reason way this is an interesting and important topic, particularly for biotechnologists. Then, we estimate all the quantities of the phylogenetic network, and show some connections with the simulation results.

Chapter 7 presents a more flexible and general algorithm for the cases where the reticulation events are many, and hence the tree topologies are not easily enumerated, in order to avoid exploring the entire space of tree topologies. In order to restrict the set of reticulations, since many of them would contribute little to the likelihood of the data, a sampling scheme based on a variable selection method is used. Specifically, we first describe the concept of this selection method in regression models, and then adapt it to our framework by turning the problem from a variable selection setting into a tree topology selection setting. Finally, we show the performance of the algorithm on simulated data, and apply it to the ribosomal protein gene *rps11* data.

Chapter 8 ends with a final summary on the study and with recommendations for future research.

# Chapter 2

# Preliminaries

## 2.1 Introduction

This preliminary chapter introduces some notation and definitions which are essential tools in later chapters, and describes some results which are used throughout the thesis. In particular here we recall some basic concepts of phylogenetics and related statistical tools such as Markov models and likelihood function of a phylogenetic trees. The material covered here, except for the phylogenetic networks, is fairly standard and can be found in one form or another in most phylogenetic texts. I wish to acknowledge two books, however, which have served as basic references: the excellent book by Felsenstein (2004), Inferring Phylogenies, as well as the more concise Yang (2006), Computational Molecular Evolution.

## 2.2 Phylogenetic trees

Phylogenetics is the study of evolutionary relationships among biological entities based on the analysis of biomolecular data such as DNA, RNA, or amino-acids. We shall concentrate on DNA sequences although the methods can be applied to amino acid sequences of proteins as well. A DNA sequence is made up of four nucleotides containing the bases adenine (A), guanine (G), cytosine (C), and thymine (T), arranged in sequences that are unique for each species. The most convenient way of presenting relationships among a group of organisms is through phylogenetic trees.

A phylogenetic tree is a DAG whose topology conveys information about the evolutionary variation among taxa (Semple and Steel, 2003). A DAG is a directed graph that does not contain paths starting and ending at the same

node. An example of phylogenetic tree is illustrated in Figure 2-1 where leaf nodes, placed at the bottom of the tree, represent present-day species, while nonleaf nodes represent ancestral species which no longer exist with the most recent common ancestor placed at the top (root) of the tree. The edge lengths indicate the amount of genetic divergence.

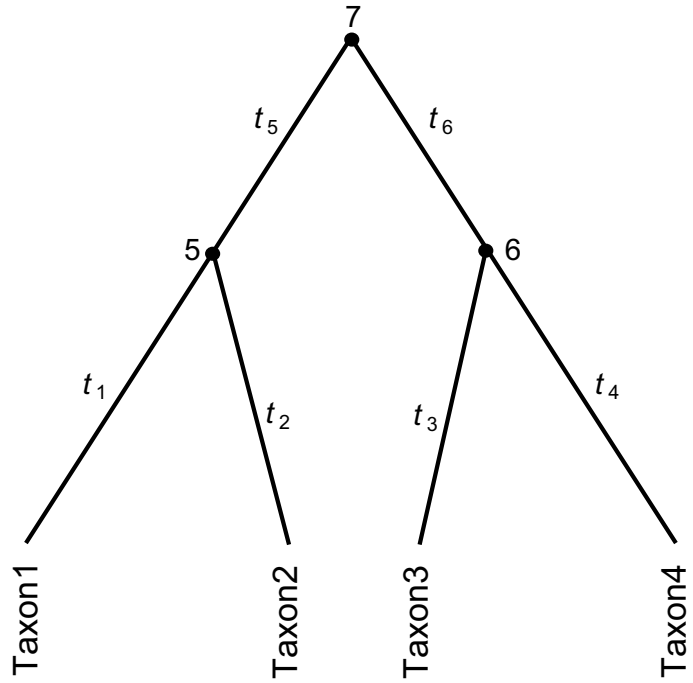



Figure 2-1: An example of phylogenetic tree $T$ of four extant species. $5$ and $6$ represent internal nodes (extinct species) and $7$ the root (the most recent common ancestor of all the taxa). Edge lengths $t_1$-$t_6$ are measured by the expected number of substitutions along the edge.

Formally, let $T = (V, E)$ be a tree, where $V$ and $E$ are the tree nodes and tree edges, respectively, and let $F(T)$ denote its leaf set and $I(T)$ its internal nodes. Additionally, let $\chi$ be a set of taxa (species). Then, $T$ is a phylogenetic tree over $\chi$ if there is a bijection between $\chi$ and $F(T)$. A tree $T$ is said to be rooted if the set of edges $E$ is directed and there is a single distinguished internal node with in-degree 0 and out-degree 2 (the most recent common ancestor). Nodes with in-degree 1 and out-degree 2 correspond to extinct species and nodes with in-degree 1 and out-degree 0 correspond to the extant species. With each edge $e \in E$ an edge length $t$, indicating the amount of evolution along the edge, expressed in terms of the average number of mutations per site, and a substitution probability $P(t)$, indicating the probability of observing different states at the two endpoints of $t$, can be associated.

Many algorithms have been designed for the inference of phylogenetic trees, by modelling the sequence of evolution by a Markov process (e.g. Felsenstein, 1981; Yang and Rannala, 1997; Mau *et al*., 1999; Larget and Simon, 1999). However, in the presence of reticulation events (see next chapter for details), it is unrealistic that the history of a set of taxa can be explained by a phylogenetic tree. In fact, in these circumstances phylogenetic networks are a more flexible tool to model biomolecular data. The next section introduces the reader to the definition of phylogenetic network.

## 2.3 Phylogenetic networks

Phylogenetic networks are a generalisation of phylogenetic trees. They allow us to model evolutionary history of a set of species in the presence of biological events that are not consistent with tree-like evolution. As extensively shown by Moret *et al.* (2004) they can be represented by DAGs. Here we consider the definition of phylogenetic network given by Moret *et al.* (2004). Notice, however, that other definitions are present in the literature (see Huson and Bryant 2006).

A phylogenetic network $N = N(T) = (V', E')$ over the taxa set $\chi$ is derived from a rooted tree $T = (V, E)$ by adding a set $R$ of reticulation edges to $T$, where each $r \in R$ is added as follows: (1) split an edge $e \in E$ by adding a new node, $v_e$, s.t. the lengths of the newly created edges sum to the length of $e$; (2) split an edge $e' \in E$ by adding new node, $v_{e'}$ (again by preserving lengths); and (3) finally, add a directed reticulation edge $r$ from $v_e$ to $v_{e'}$. Notice that the substitution probability (and hence the length) of $r$ is zero as these events are instantaneous in time. The resulting network is a rooted directed acyclic graph. Figure 2-2 shows an example of phylogenetic network obtained by adding the edge (5,6) to the underlying species tree shown in Figure 2-3a. Assuming that the species tree is available is not completely unreasonable. In fact for many taxa the underlying organismal tree can be inferred with high degree of probability or confidence. The tree in Figure 2-3a models the evolution of all genetic material that is vertically inherited from the species tree, whereas the tree in Figure 2-3b models the evolution of horizontally transferred genetic material. Denote by $\mathbf{T}(N) = (T_1, \ldots, T_K)$ the set of all trees contained inside network $N$. Each such tree is obtained by the following two steps: (1) for each node with in-degree 2, remove one of the incoming edges, and then (2) for every node $w$ of in-degree and out-degree 1, whose parent is $u$ and child $v$,

remove node $w$ and its two adjacent edges, and add a new edge from $u$ to $v$ (while summing the edge lengths). For example, the set $\mathbf{T}(N)$ of the network $N$ in Figure 2-2 contains only the two trees that are shown in Figure 2-3. In general, a network $N$ with $R$ reticulation events can induce up to $K = 2^R$ trees. Note that sometimes the number of possible trees in the network is less than $2^R$ (see section 6.3). Finally, as shown in Figures 2-2 and 2-3, there may be common edge lengths among the trees induced by the network.

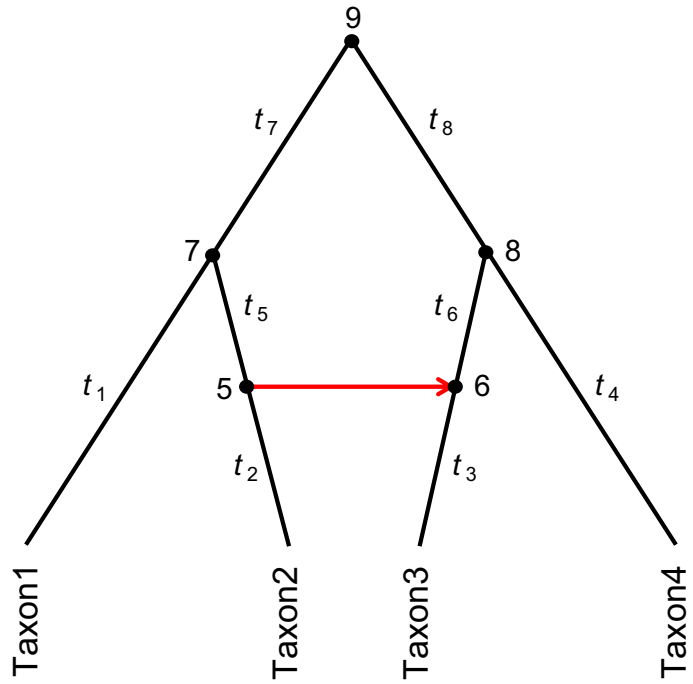
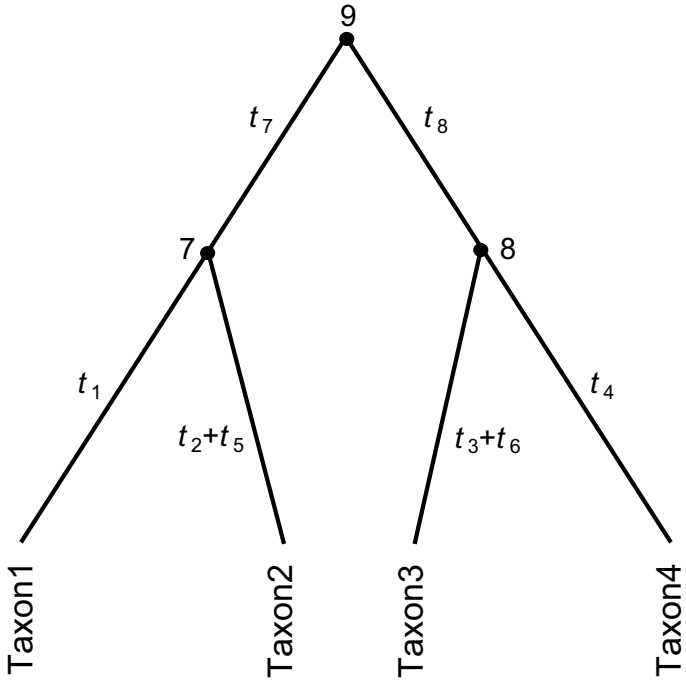


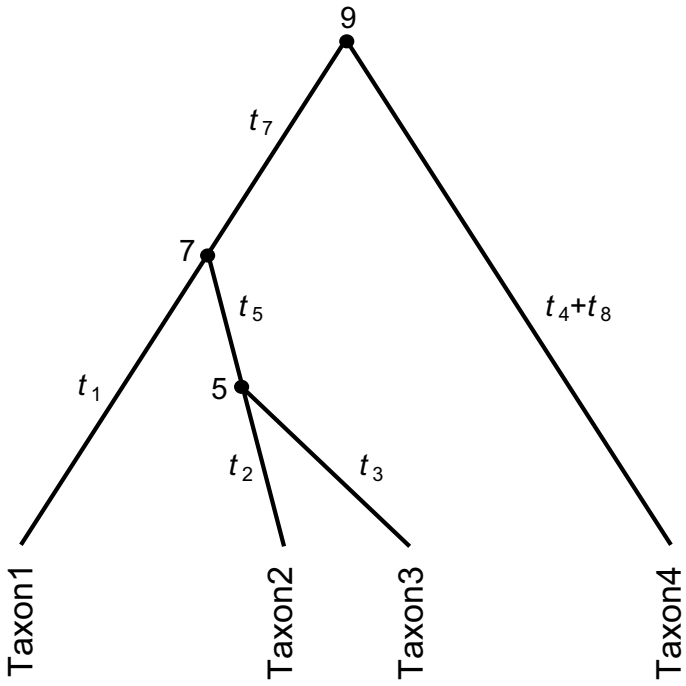

Figure 2-2: An example of phylogenetic network $N$ of four taxa with one reticulation edge ($R = 1$). (5-8) represent internal nodes and 9 the root. Edge lengths $t_1$-$t_8$ are measured by the expected number of substitutions along the edge.

## 2.4 Markov models of sequence evolution

As we will shortly see, phylogenetic models, as considered here, define a stochastic process of substitution that operates independently at each site in DNA sequences. In the assumed process, a character is first drawn at random from the background distribution and assigned to the root of the tree; character substitutions then occur randomly along the trees branches, from root to leaves. The characters that remain at the leaves when the process has com-




(a) The underlying species tree $T_1$, that is, the tree that does not include the reticulation edge (5,6).




(b) The horizontally transferred gene tree $T_2$, that is, the tree that includes the reticulation edge (5,6).

Figure 2-3: Trees induced by the network $N$.

pleted define an alignment column. Thus, a phylogenetic model induces a distribution over alignment columns having a correlation structure that reflects the phylogeny and substitution process. Notice that here we just describe the stochastic process of substitution for one single site.

Consider the representation of site evolution in Figure 2-4. Over a time period $t$, the state $G$ at the site is replaced by the state $T$. There are three random mutation events that are randomly distributed through the time period. One of these is redundant, with $G$ being replaced by $G$. These redundant mutations are considered more for mathematical convenience than anything else. The mutations from $G$ to $A$ and from $A$ to $T$ are said to be silent. The change to $A$ is not observed, only the beginning and end states. Let $\Sigma$ denote the set of states (e.g. for nucleotide data, $|\Sigma| = 4$). The mutation events occur accord-

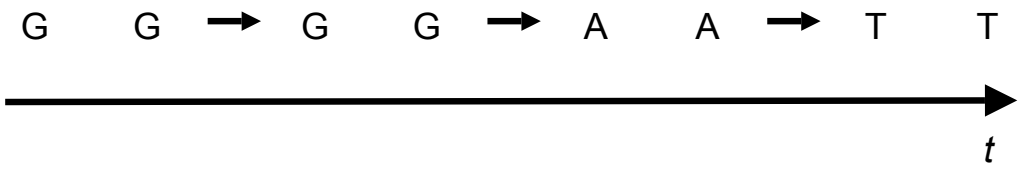



Figure 2-4: Redundant and hidden mutations. Over time $t$, the site has a redundant mutation, followed by a mutation to A and then to T. The mutation to A is not detectable (silent mutation). The number of the events is modelled by a Poisson process.

ing to a continuous time Markov chain with state set $\Sigma$. The number of these events has a Poisson distribution: the probability of $k$ mutation events is

$$P(k) = \frac{(\mu t)^k e^{-\mu t}}{k!}.$$

Here $\mu$ is the rate of these events, so that the expected number of events in time $t$ is $\mu t$. When there is a mutation event, we let $M_{xy}, (x, y \in \Sigma)$ denote the probability of changing to state $y$ given that the site was in state $x$. Since redundant mutations are allowed, $M_{xx} > 0$. Putting everything together, the probability of ending in state $y$ after time $t$ given that the site started in state $x$ is given by the $xy^{th}$ element of $\mathbf{P}(t)$, where $\mathbf{P}(t)$ is the matrix valued function

$$\mathbf{P}(t) = \sum_{k=0}^{\infty} \mathbf{M}^k \frac{(\mu t)^k e^{-\mu t}}{k!} = e^{-\mu t} \sum_{k=0}^{\infty} (\mathbf{M}^k) \frac{(\mu t)^k}{k!} \tag{2.1}$$

and $\mathbf{M}$ is the matrix whose entry is $M_{xy}$. The probability matrix, $\mathbf{P}(t)$, is known as the transition probability matrix. This formula just expresses the probabili-

ties of change summed over the possible values of $k$, the number of mutation events. The sum in (2.1) has the same form as the series approximation of the matrix exponentiation

$$e^A = \sum_{k=0}^{\infty} \frac{A^k}{k!}$$

and so (2.1) can be rewritten as

$$\mathbf{P}(t) = e^{-\mu t} e^{\mathbf{M}\mu t} = e^{-\mu t \mathbf{I}} e^{\mathbf{M}\mu t} = e^{(\mathbf{M}-\mathbf{I})\mu t} = e^{\mathbf{Q}\mu t} \tag{2.2}$$

where $\mathbf{I}$ is the $4 \times 4$ identity matrix and $\mathbf{Q} = \mathbf{M} - \mathbf{I}$ is the instantaneous substitution rate matrix. So, as shown in 2.2, the transition probabilities $\mathbf{P}(t)$ over some time period $t$ can be obtained by exponentiating the $\mathbf{Q}$ matrix. There is a standard trick to compute it.

First, diagonalise the matrix $\mathbf{Q}$ as

$$\mathbf{Q} = \mathbf{S}\mathbf{\Lambda}\mathbf{S}^{-1}$$

where $\mathbf{S}$ is a matrix whose columns are the right eigenvectors of $\mathbf{Q}$ and $\mathbf{\Lambda}$ is a diagonal matrix containing the eigenvalues ($\lambda_x$). For any integer $k$ we have that

$$\begin{aligned} (\mathbf{Q})^k &= \left(\mathbf{S}\mathbf{\Lambda}\mathbf{S}^{-1}\right)\left(\mathbf{S}\mathbf{\Lambda}\mathbf{S}^{-1}\right)\ldots\left(\mathbf{S}\mathbf{\Lambda}\mathbf{S}^{-1}\right) \\ &= \mathbf{S}\left(\mathbf{\Lambda}\right)^k \mathbf{S}^{-1}. \end{aligned}$$

Taking the powers of diagonal matrices is just a matter of taking the powers of its entries. It follows that

$$e^{\mathbf{Q}\mu t} = e^{\mu t}\mathbf{S}e^{\mathbf{\Lambda}}\mathbf{S}^{-1},$$

where $e^{\mathbf{\Lambda}}$ is a diagonal matrix and, for each $x$, $\left(e^{\mathbf{\Lambda}}\right)_{xx} = e^{\mathbf{\Lambda}_{xx}}$. The transition probability matrix is known in closed form for simple instantaneous rate matrices but not for the more complex ones. In the latter cases it is common to use routines for numerically determining the eigenvalues and eigenvectors to derive the transition probabilities. However, the eigenvalue decomposition methods can occasionally be numerically inaccurate, in which case Golub and van Loan (1996) recommend using the Padé approximation. The Padé approximation to $e^{\mathbf{Q}}$ is

$$e^{\mathbf{Q}} \approx R\left(\mathbf{Q}\right),$$

with

$$R_{pq}(\mathbf{Q}) = (D_{pq}(\mathbf{Q}))^{-1} N_{pq}(\mathbf{Q})$$

where

$$D_{pq}(\mathbf{Q}) = \sum_{j=1}^{p} \frac{(p+q-j)!p!}{(p+q)!j!\,(p-j)!}\mathbf{Q}^j$$

and

$$N_{pq}(\mathbf{Q}) = \sum_{j=1}^{q} \frac{(p+q-j)!q!}{(p+q)!j!\,(q-j)!}\mathbf{Q}^j.$$

Several functions in R perform matrix exponentiation. Here we use the function `expm` in the package `expm` with default parameters $p = q = 8$.

As an example, consider the general time reversible model (GTR) which was first described by Tavaré (1986). This is the most general model in that it allows different nucleotides to be substituted at different rates as well as different equilibrium frequencies. Assuming that the states in $\Sigma$ are ordered $A, C, G, T$, the model defined in terms of its rate matrix is

$$\mathbf{Q} = \begin{bmatrix} -Q_{AA} & r_{AC}\pi_C & r_{AG}\pi_G & r_{AT}\pi_T \\ r_{AC}\pi_A & -Q_{CC} & r_{CG}\pi_G & r_{CT}\pi_T \\ r_{AG}\pi_A & r_{CG}\pi_C & -Q_{GG} & r_{GT}\pi_T \\ r_{AT}\pi_A & r_{AC}\pi_C & r_{GT}\pi_G & -Q_{TT} \end{bmatrix} \tag{2.3}$$

where $\mathbf{r} = \{r_{AC}, r_{AG}, \ldots, r_{GT}\}$ is the vector of the rate parameters and $\boldsymbol{\pi} = \{\pi_A, \pi_C, \pi_G, \pi_T\}$ the vector of stationary frequencies. The entries of the matrix $\mathbf{Q}$ are given by $Q_{xy} = r_{xy}\pi_y$,($x, y \in \Sigma$, $x \neq y$). Each row sum of the matrix is zero and hence $Q_{xx} = \sum_{y\in\Sigma, y\neq x} Q_{xy}$. The entries of $\mathbf{P}(t)$, $P_{xy}(t)$, satisfy $P_{xy}(t) \geq 0$ and $\sum_{y\in\Sigma} P_{xy}(t) = 1$. In addition the transition probabilities satisfy the Chapman-Kolmogorov equations:

$$\sum_{k\in\Sigma} P_{xk}(s)P_{ky}(t) = P_{xy}(s+t),\ \forall s, t \geq 0$$

and the initial condition $P_{xy}(0) = 1$ if $x = y$ and $P_{xy}(0) = 0$ if $x \neq y$. To understand the Chapman-Kolmogorov equation think of $\Sigma = \{A, C, G, T\}$ and the process being in state $A$ reaching state $T$ in time $s+t$. The transition probability $P_{AT}(s+t)$ is

$$
\begin{aligned}
P_{AT}(s+t) &= P_{AA}(t)\,P_{AT}(s+t) \\
&+ P_{AC}(t)\,P_{CT}(s+t) \\
&+ P_{AG}(t)\,P_{GT}(s+t) \\
&+ P_{AT}(t)\,P_{TT}(s+t) \\
&= \sum_{k\in\Sigma} P_{Ak}(s)P_{kT}(t).
\end{aligned}
$$

The substitution process is also assumed to be time reversible, that is

$$\pi_x P_{xy}(t) = \pi_y P_{yx}(t).$$

Note that $\pi_x$ is the proportion of time the Markov chain spends in state $x$, and $\pi_x P_{xy}(t)$ is the amount of flow from state $x$ to state $y$, while $\pi_y P_{yx}(t)$ is the flow in the opposite direction. This equation is known as detailed balance condition and means that the flow between any two states in the opposite direction is the same. There is no biological reason to expect the substitution process to be reversible, so reversibility is a mathematical convenience (Yang, 2006).
By introducing extra constraints on the nucleotide frequencies $(\pi_x, x \in \Sigma)$ and/or on the relative substitution rates $(r_{xy}, x, y \in \Sigma, x \neq y)$, different substitution models can be obtained. The simplest one, described by Jukes and Cantor (1969), assumes that the stationary frequencies of all nucleotides and all the relative substitution rates are equal $(\pi_A = \ldots = \pi_T)$ and $(r_{AC} = \ldots = r_{GT})$, and hence the rate matrix simplifies to

$$\mathbf{Q} = \begin{bmatrix} -3r\pi & r\pi & r\pi & r\pi \\ r\pi & -3r\pi & r\pi & r\pi \\ r\pi & r\pi & -3r\pi & r\pi \\ r\pi & r\pi & r\pi & -3r\pi \end{bmatrix} \tag{2.4}$$

where $r = r_{AC} = \ldots = r_{GT}$ and $\pi = \pi_A = \ldots = \pi_T$.

It is important to mention that the rate $\mu$ and time $t$ occur in the transition probabilities only in the form of a product $\mu t$. With no external information about either the time or the rate, we can estimate only the distance as given by the product of these two quantities, but not them individually. This means the absolute rate of change cannot generally be estimated but only the amount of change. Therefore, typically the branch lengths in phylogenetic trees are

measured in terms of the amount of change. In particular, a branch length unit corresponds to one expected change per site at stationarity. This requires the scaling of the $\mathbf{Q}$ matrix by a standardising factor $\mu$, namely that:

$$\mu \sum_{x \in \Sigma} -Q_{xx}\pi_x = 1.$$

## 2.5 Likelihood function of a phylogenetic tree

Consider an alignment $\mathbf{D}$ of $p$ DNA sequences, $n$ nucleotides long. Let each column in the alignment be represented by $\mathbf{d}_i$, where the subscript $i$ represents the site, $1 \leq i \leq n$. Hence $\mathbf{d}_i$ is a column vector of length $p$ containing the nucleotides $\{A, C, G, T\}$ at the $i^{th}$ site of the alignment, and $\mathbf{D} = (\mathbf{d}_1, \ldots, \mathbf{d}_n)$. Each column of the alignment is a realisation of a continuous time Markov evolutionary process on phylogenetic tree topology $T$, with branch lengths $\mathbf{t}$, where $\mathbf{t}$ are expressed in terms of amount of evolutionary change. This process is governed by the infinitesimal rate matrix $\mathbf{Q}$. Denote the set of unknown parameters of the phylogenetic tree (the branch lengths $\mathbf{t}$, the parameters of the nucleotide substitution model $\boldsymbol{\pi}$ and $\mathbf{r}$, and the tree topology $T$) by $\boldsymbol{\theta}$ and the likelihood function for the observed DNA sequences from a Markov evolutionary process by $L(\mathbf{D}|\boldsymbol{\theta})$. Because of the assumption of independent evolution among sites, the overall likelihood $L(\mathbf{D}|\boldsymbol{\theta})$ of the aligned sequences $\mathbf{D}$ given the parameters $\boldsymbol{\theta}$ is obtained by the product of the probabilities of data at individual sites $P(\mathbf{d}_i|\boldsymbol{\theta})$

$$L(\mathbf{D}|\boldsymbol{\theta}) = L(\mathbf{D}|\mathbf{t}, \boldsymbol{\pi}, \mathbf{r}, T) = \prod_{i=1}^{n} P(\mathbf{d}_i|\boldsymbol{\theta}). \tag{2.5}$$

Equivalently the log likelihood is a sum over sites in the sequence:

$$\ell = \log(L) = \sum_{i=1}^{n} \log\{P(\mathbf{d}_i|\boldsymbol{\theta})\}.$$

Felsenstein (1981) introduced a dynamic programming procedure, called the pruning algorithm, in order to compute the likelihood function $L(\mathbf{D}|\boldsymbol{\theta})$ in a fast and efficient way. Its essence is to calculate successively probabilities of data on many subtrees. Let $v$ be an internal node of the tree, and let $L_i^v(x)$, $x \in \Sigma$, denote the probability of observing data at the tips that are descendants

of node $v$, given that the nucleotide at node $v$ is $x$, that is:

$$L_i^v(x) = P\{\mathbf{d}_i^v | \boldsymbol{\theta}, \widehat{\mathrm{d}}_i(v) = x\},$$

where $\mathbf{d}_i^v$ is the restriction of $\mathbf{d}_i$ to the descendants of node $v$ and $\widehat{\mathrm{d}}_i(v)$ is the ancestral state for site $i$ at node $v$. In other words, the value $L_i^v(x)$ is the probability at site $i$ for the subtree underlying node $v$, conditional on state $x$ at $v$. In the literature, the conditional probability $L_i^v(x)$ is often confusingly referred to as either the 'partial likelihood' or 'conditional likelihood'.

The probability of the complete character $\mathbf{d}_i$ can be expressed as:

$$P(\mathbf{d}_i|\boldsymbol{\theta}) = \sum_{x \in \Sigma} P\{\widehat{\mathrm{d}}_i(v_0) = x\} L_i^{root}(x),$$

where $v_0$ is the root node. $P\{\widehat{\mathrm{d}}_i(v_0) = x\}$ is the (prior) probability that the nucleotide at the root is $x$, and is usually given by the equilibrium frequency of the nucleotide under the model ($\pi_x, x \in \Sigma$).

The function $L_i^v(x)$ satisfies the recurrence:

$$L_i^v(x) = \left( \sum_{y \in \Sigma} P_{xy}(t_1) L_i^{v_1}(y) \right) \left( \sum_{y \in \Sigma} P_{xy}(t_2) L_i^{v_2}(y) \right) \tag{2.6}$$

for all internal nodes $v$, where $v_1$ and $v_2$ are the children of $v$ and $t_1$, $t_2$ are the lengths of the branches connecting them to $v$. Equation (2.6) results from the independence of the processes in the two subtrees below node $v$. For leaf $l$, we have

$$L_i^l(x) = \begin{cases} 1 & \text{if} \quad \mathbf{d}_i(l) = x \\ 0 & \text{if} \quad \mathbf{d}_i(l) \neq x \end{cases}.$$

Note that (2.6) can be easily extended to nodes $v$ with more than two children. The transition probabilities $P_{xy}(t_1)$ and $P_{xy}(t_2)$ are determined from (2.2) which requires, as observed above, the diagonalisation of the rate matrix $\mathbf{Q}$.

**Savings on computation**

The pruning algorithm is the major time-saver. Some other obvious savings may also be made:

1. The same transition-probability matrix is used for all the sites in the sequence and may be calculated only once for each branch.

2. If two sites have the same data, the probabilities of observing them (pro-

vided the share a common tree) will be the same and need to be calculated only once. Collapsing sites into site patterns thus leads to a saving in computation, especially if the sequences are highly similar so that many sites have identical patterns.

## 2.6 Hidden Markov models

In Section 2.4 we discussed probabilistic models that consider the way substitutions occur through evolutionary history at each site of a sequence, and in Section 2.5 we assumed that evolution is independent among sites. Of course this assumption can be too restrictive in practice, as adjacent sites are not necessarily independent for example with respect to reticulation events. HMMs are probabilistic models that consider not only how substitutions occur along the branches of a phylogenetic tree at each site of a sequence, but also the way this process changes from one site to the next. By treating molecular evolution as a combination of two Markov processes (one that operates in the dimension of space, along sequences, and one that operates in the dimension of time, along the branches of a phylogeny) these models allow aspects of both sequence structure and sequence evolution to be captured.

Informally speaking, with HMMs, a sequence of observations $\mathbf{D}$ is available to be analysed, but the sequence of states by which the observations were generated is 'hidden' (hence the name hidden Markov model). More precisely, an HMM, can be described as a model in which a sequence of observations $(\mathbf{d}_1, \ldots, \mathbf{d}_i, \ldots, \mathbf{d}_n)$, is modelled by specifying a probabilistic relation between observations and a sequence of hidden states $S_i$, and a Markov transition structure linking the hidden states. The model assumes two sets of conditional independence relations: that $\mathbf{d}_i$ is independent of all other observations and states given $S_i$, and that $S_i$ is independent of $S_1 \ldots S_{i-2}$ given $S_{i-1}$ (the first-order Markov property). Using these independence relations, the joint probability for the sequence of states and observations can be factored as

$$P(\mathbf{S}, \mathbf{D}) = P(S_1)P(\mathbf{d}_1|S_1) \prod_{i=2}^{n} P(S_i|S_{i-1})\, P(\mathbf{d}_i|S_i)\,. \tag{2.7}$$

The conditional independencies specified by equation (2.7) can be expressed graphically in the form of Figure 2-5. The state is a single multinomial random variable that can take one of $K$ discrete values, $S_i \in \{1, \ldots, K\}$. $P(S_i|S_{i-1})$ are the state transition probabilities and $P(\mathbf{d}_i|S_i)$ the probabilities of the data

given the hidden states. For a rigorous treatment of HMM in phylogenetics see Siepel and Haussler (2005).

Another possibility would be to use Markov random fields (MRFs). The key idea behind MRFs is that the distribution of the process at a particular location depends only on the values of the process at neighboring locations. In this sense, it is Markovian; points outside the neighborhood are conditionally independent given the neighborhood. One of the degrees of flexibility in the model is the specification of the neighborhood. Often, only the immediately adjacent points will constitute the definition of the neighborhood (first-order neighborhoods). However, more extended neighborhoods are easily incorporated into the structure if desired. On a regular grid in two dimensions, this would be the four lattice points (or grid cells) that are horizontally or vertically adjacent. In one dimension, there are only two neighbors. First order MRFs might be a more realistic tool as compared to first order HMMs, in that they would allow us to model dependencies between sites accounting for both neighbors. Future research could explore this possibility; we refer the reader to Rue and Held (2005) and Besag (1974) for a detailed and more general introduction.

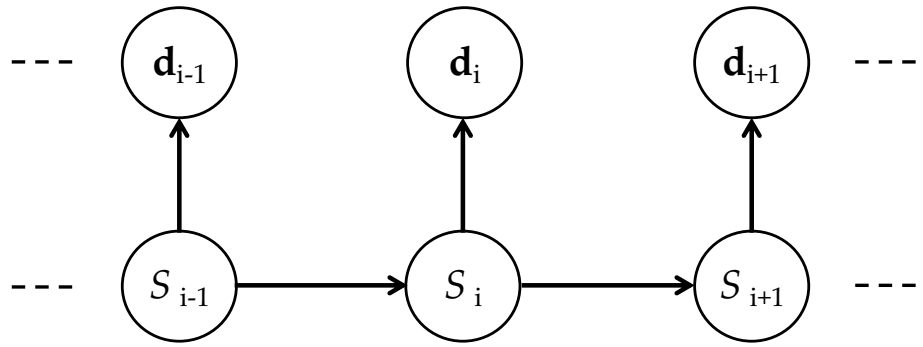



Figure 2-5: DAG representing the dependence structure of (2.7). The $\mathbf{d}_i$ represent the columns in the DNA data and the $S_i$ hidden states.

# Chapter 3

# Phylogenetic networks in general

## 3.1 Introduction

We begin by providing some background on reticulation events, discussing the consequences of failing to model reticulation events, and giving some references of works in related problems. Then, we describe the biology underlying reticulation events. In particular we review the concept of horizontal gene transfer (HGT) and hybrid speciation (HS). We also look at the existing methods to model these events in the literature, specifically the *naive* ML estimation, ML with HMMs, and MP. We conclude by discussing advantages and limitations of these methods and discuss an alternative approach for this area of research.

## 3.2 Background

As previously discussed, phylogenetic trees, which are the most used tool for representing evolutionary relationships among species, may oversimplify our view of evolution as they cannot account for reticulate evolutionary events (i.e. exchange of genetic material between two or more taxa), such as horizontal gene transfer (HGT) and hybrid speciation (HS)(see Linder *et al.*, 2004 and references therein). These are among the fundamental processes creating diversity at the gene level, particularly among bacteria (see Heuer and Smalla, 2007) and plants (Bergthorsson *et al.*, 2003; Bergthorsson *et al.*, 2004; Linder and Rieseberg, 2004; Richardson and Palmer, 2007), and causing antibiotic resistance genes in the environment (see Martínez *et al.*, 2007 and references therein) which is a major factor that limits the effectiveness of antibiotics. HGT and HS are potential confounding factors in inferring phylogenetic trees;

failure to consider these events might result in incorrectly inferred phylogenies. For example, given two distantly related species that have exchanged a gene (or part of it), a phylogenetic tree including those taxa will show them to be closely related because that gene (or part of it) is the same, even though most other genes (or part of them) are dissimilar. This is illustrated in Figure 3-1. In this respect, accounting for reticulation events is crucial as it allows for improved phylogenetic inference. For these reasons, it is often ideal to use alternative tools to infer robust phylogenies.

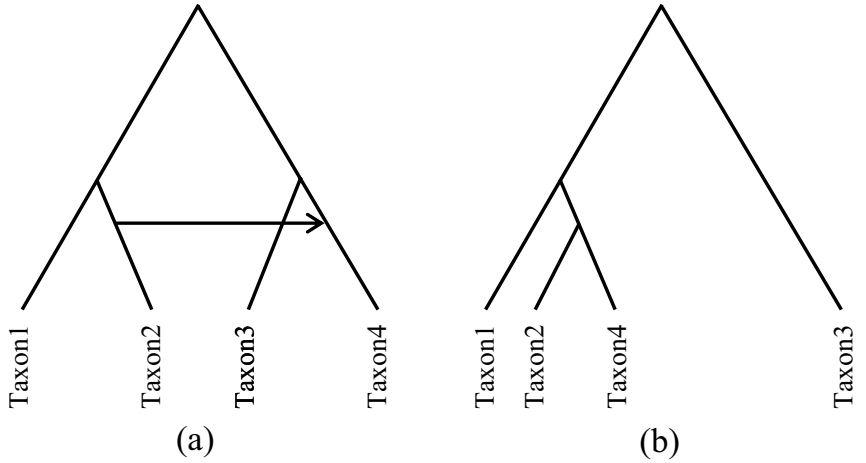



Figure 3-1: An example of an incorrectly inferred phylogenetic tree due to a reticulation event: (a) contains the underlying species tree of four taxa with taxon 2 transferring genetic material to taxon 4. (b) is the inferred phylogenetic tree which wrongly relates taxon 2 and taxon 4 because of the unmodelled reticulation event.

In recent years many researchers have introduced a number of tools to model evolutionary relationships among taxa in the presence of reticulations. Most of these methods fall under the general umbrella of phylogenetic networks. Broadly, two categories of phylogenetic networks can be distinguished. These are *split networks* and *reticulate networks*.

Split networks are representations based on bipartitions (splits) that capture conflicting signals in the data due to various factors, not necessarily reticulation events. In such a network, parallel edges, rather than single branches, are used to represents the splits computed from the data. To be able to accommodate incompatible splits, it is often necessary that a split network contains nodes that do not represent ancestral species. Thus, split networks provide only an implicit representation of evolutionary relationships. In fact, they are largely employed to display incompatibilities in data. However, this is of-

ten not the scope of reticulate analysis. As pointed out by Huson and Bryant (2006) split networks provide only an implicit representation of evolutionary history and do not construct a model for phylogenetic inference. Rather, they construct a model for graphical representation of data sets when trees fail to be the appropriate model. This is why, split networks are generally employed as a tool to represent and explore data in the same way as a scatter plot can be used to explore the relationship between two variables. Some of the better known and used networks belonging to this class of methods are: median networks, consensus networks, split decomposition and neighbour-net methods. A very brief overview of these networks is given.

In the median network method (Bandelt et al., 1995; Bandelt et al., 2000) biomolecular sequences are first converted into binary data and then, constant sites are eliminated. Each split is encoded as a binary character with states 0 and 1. Sites that support the same split are grouped in one character, which is weighted by the number of sites grouped. Median vectors are calculated for each triplet of vectors until the median network is finished. Such networks can become very complex due to the presence of high dimensional hypercubes. Fortunately, there exist techniques that can reduce this complexity preserving the underlying phylogenetic signals (see for example Bandelt et al., 1995).

There are many situations in phylogenetics where the methods employed produce a collection of trees. For instance, the trees might be the result of a bootstrap analysis, samples from a posterior distribution, or might come from a multigene analysis. Large collections of trees can be difficult to interpret and draw conclusions from. Therefore, it is common practice to summarise the information contained in all of the trees by using a consensus tree. However, this practice suffers from a limitation: by summarising all of the given trees by a single output tree, information about conflicting hypotheses can be lost. Common methods for computing a consensus tree are the strict consensus method, which outputs the tree displaying only those bipartitions that appear in all the source trees, and the majority consensus method, which outputs the tree displaying only those splits that appear in more than half of the input trees. These two methods can be viewed as members of a one parameter family consensus method which associates a split system to a collection of phylogenetic trees consisting of those splits that are displayed by more than a given proportion of the trees. In case the proportion is less than one half, it may no longer be possible to associate a tree to the split system, as it may contain some pairs of incompatible splits. However it is still possible to represent the split system by a phylogenetic network. Such a network is called a consensus network

(Holland and Moulton, 2003).

There also exist a number of methods that generate incompatible splits directly from a distance matrix. The split decomposition (Bandelt and Dress, 1992) is one of those. It takes as input a distance matrix on a set of taxa and produces a set of weighted splits, where the sum of weights of all splits that separate two taxa is an approximation of the given distance. This method has the nice property that it produces a tree, whenever the distance matrix is a tree, and otherwise it produces a treelike splits network that potentially displays different and conflicting signals in a given data set. Although split decomposition is useful for visualising conflicting signals in a data set, it is sensitive to noise and only has good resolution for data sets of up to about 20 taxa.

The neighbor-net method (Bryant and Moulton, 2004) is a generalisation of the tree building method neighbor-joining (NJ). It is applicable to data sets containing hundreds of taxa. NJ is a cluster algorithm based on the minimum evolution criterion. It starts with a star tree and then joins two nodes, choosing the pair to achieve the greatest reduction in tree length. A new node is then created to replace the two nodes joined, reducing the dimension of the distance matrix by one. The procedure is repeated till the tree is fully resolved. Neighbor-net has one important difference. When pairs of nodes are selected, they are not combined and replaced immediately. Instead, the method waits until a node has been paired up a second time, at which stage three linked nodes are replaced with two linked nodes. In case a node linked to two others remains, a second agglomeration and reduction is performed.

In contrast to split networks, reticulate networks give an explicit picture of evolution and can be thought of as an extension of phylogenetic trees able to directly model reticulations. Roughly speaking, they can be grouped into two categories: reticulate networks at the *population level* (which model sexual recombination) and reticulate networks at the *species level* (which model HGT and HS). As for the reticulate networks at the population level, Strimmer *et al*. (2001) and Husmeier and McGuire (2002) gave important contribution to the field. In particular, Strimmer *et al*. (2001) propose a stochastic network based on the concept of the ancestral recombination graph (Hudson, 1983; Griffiths and Marjoram, 1996) as a way to model phylogenetic networks. This is a rooted graph that provides a way to represent a linked collection of clock-like trees by a single network. They also describe how the likelihood of the data under a genealogy based on these graphs can be computed. Husmeier and McGuire (2002) model recombination at the population level in a Bayesian fashion. They present a statistical model for detecting recombina-

tion, whose objective is to accurately locate the recombination breakpoints in DNA sequence alignments. Their approach explicitly models the sequence of phylogenetic tree topologies along a multiple sequence alignment. Inference under this model is done in a Bayesian way using Markov chain Monte Carlo. The algorithm returns the site-dependent posterior probability of each tree topology, which is used for detecting recombinant regions and locating their breakpoints.

One methodology for network reconstruction at the species level has been proposed by Moret and collaborators (2004) who defined a phylogenetic network as a DAG obtained by positing a set of edges between pairs of the branches of an underlying tree (species tree) to model reticulation events (see previous chapter). Recently, three works (Jin *et al.*, 2006; Snir and Tuller, 2009; Nakhleh *et al.*, 2005) using this definition of phylogenetic network, have appeared. In particular, the first one demonstrates the potential of using ML estimation for phylogenetic network reconstruction. The second is an extension of the previous work in that ML is combined with HMMs. The third uses an MP approach for inferring evolutionary networks. All these methods have been published in advanced computational biology journals, where the arguments are not always fully developed, and hence easy to follow. Other methods for phylogenetic networks at the species level have been proposed by Suchard (2005) and Linz *et al*. (2007). However these latter works suggest approaches to estimate an overall rate of HGT, rather than reconstructing phylogenetic networks. In Section 3.4 we review the three reticulate network methods at the species level with the aim of providing an accessible introduction to this fascinating field of research. Other reviews are available in the literature (e.g. Posada and Crandall, 2001; Morrison, 2005; Huson and Bryant, 2006; Makarenkov *et al.*, 2006); the majority of them discuss split networks without distinguishing between reticulations at different levels, and do not describe methods based on the idea that a phylogenetic network can be naturally and intuitively decomposed into trees.

## 3.3 Understanding the biology behind reticulation events

Biologists indicate by the term reticulation the dependence between two or more evolutionary lineages. In fact, when reticulation occurs, two (or more) independent evolutionary lineages are combined at some biological level. Be-

cause life is organized hierarchically, reticulation can occur at three different levels: chromosomal, population, and species.

At the chromosomal level reticulation is called meiotic recombination which is a process by which two chromosomes, paired up during one phase of meiosis (process of reductional division in which the number of chromosomes per cell is cut in half) exchange some portion of their DNA. In other words, meiotic recombination occurs when two chromosomes break and then reconnect but to different end pieces.

At the population level reticulation takes the name of sexual recombination. During this process half of one parent's genes are combined with half of the other parent's genes in the offspring, which results in a gene combination that did not previously exist.

At the species level, events such as HS (two species recombine to form one new species) and HGT (one species horizontally transfers genetic material to another species) are the main causes of reticulate evolution. Figure 3-2, taken from Linder *et al*. (2004), provides an excellent and illustrative example of reticulation event at the three levels. The tree depicted in Figure 3-2a illustrates a scenario of hybrid speciation, in which species 2 and species 4 recombine to create species 3. Zooming in on a lineage of the tree gives a picture of reticulate event at the population level, as shown in Figure 3-2b. Finally, zooming in on an individual in each population, meiotic recombination can be viewed, as illustrated in Figure 3-2c. Since the aim of the chapter is to review existing methods for modelling reticulation events at the species level, the next section will describe in more detail the concept of horizontal gene transfer and hybrid speciation. For a formal and comprehensive discussion of reticulation events at the chromosomal and population level, the reader is referred to Linder and colleagues (2004), and references therein.

### 3.3.1 Reticulation at the species level

#### Horizontal gene transfer

Horizontal (also called lateral) gene transfer occurs when genetic material is horizontally transferred from one species to another (see Figure 3-3a where taxon 2 horizontally transfers genetic material to taxon 3). In an evolutionary scenario involving horizontal transfer, some sites are inherited through lateral transfer from another species (Figure 3-3c), while all others are inherited from the parent (Figure 3-3b). Thus, each site evolves down one of the trees con-

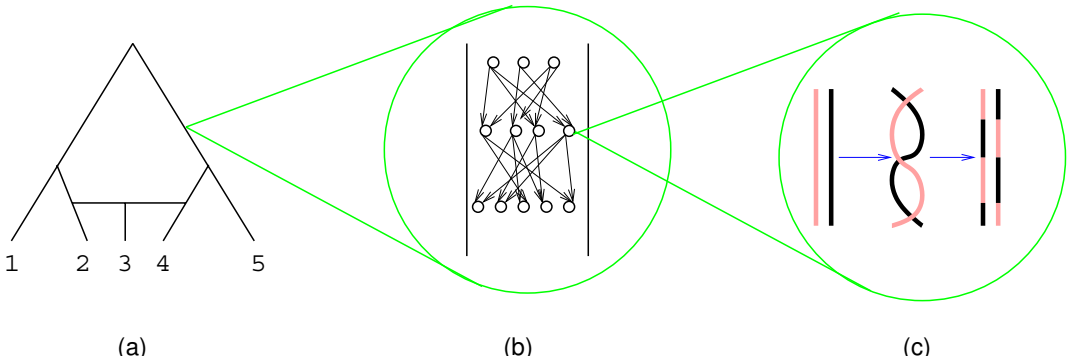



Figure 3-2: Reticulation at the (a) species, (b) population, and (c) chromosomal level.

tained inside the network. HGTs are extremely frequent in bacteria, although this view has recently been challenged (see Linder *et al.*, 2004 and references therein). There are three common mechanisms for lateral gene transfer in bacteria: *transformation*, that is the genetic alteration of a cell resulting from the introduction, uptake and expression of foreign genetic material; *conjugation*, a process in which a living cell transfers genetic material through cell-to-cell contact; *transduction* the process in which bacterial DNA is moved from one bacterium to another by a bacterial virus.

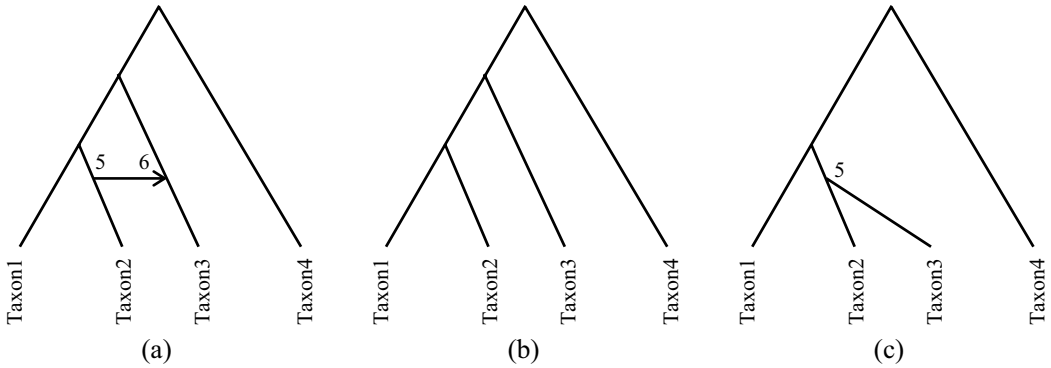



Figure 3-3: An example of HGT. The phylogenetic network $N$ of four taxa with one HGT in (a) and the two possible trees $\mathbf{T}(N) = (T_1, T_2)$ induced by $N$ in (b) and (c). In particular (b) contains the underlying species tree $T_1$, that is the tree that does not include the HGT edge and (c) the horizontally transferred gene tree $T_2$, that is the tree that includes the HGT edge.

**Hybrid speciation**

In hybrid speciation, two lineages recombine to create a new species (see Figure 3-4a where taxon 1 and taxon 3 recobine giving rise to taxon 2). This non-treelike event is very common in some group of organisms (plants, fish, fungi for example) and is virtually absent in others (mammals and most arthropods). Similarly to HGT, in an evolutionary scenario of HS, certain sites are inherited from the parent (Figure 3-4b), while others are inherited through hybrid speciation (Figure 3-4c). We can distinguish different ways of HS: *diploid hybridization*, in which the new species inherits one of the two homologs (i.e., chromosomes having the same genes at the same loci but possibly different alleles) for each chromosome from each of its two parents so that the new species has the same number of chromosomes as its parents; *polyploid hybridization*, in which the new species inherits the two homologs of each chromosome from both parents so that the new species has the sum of the numbers of chromosomes of its parents. Under this last heading, we can further distinguish *allopolyploidization*, in which two lineages hybridize to create a new species whose ploidy level (which refers to the number of complete sets of chromosomes in each cell) is the sum of the ploidy levels of its two parents, and *autopolyploidization*, a regular speciation event that does not involve hybridization, but which doubles the ploidy level of the newly created lineage.

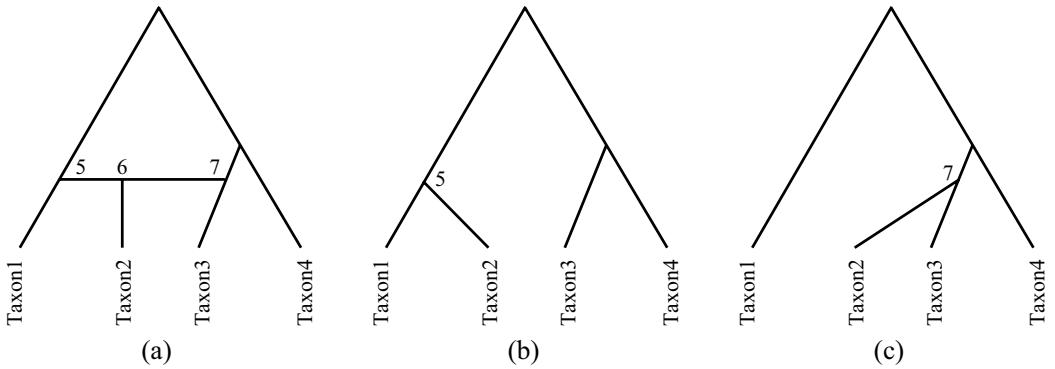



Figure 3-4: An example of HS. The phylogenetic network $N$ of four taxa with one HS event in (a) and the two possible trees $\mathbf{T}(N) = (T_1, T_2)$ induced by $N$ in (b) and (c). In particular (b) contains the underlying species tree $T_1$, that is the tree that does not include the hybrid speciation edge and (c) the horizontally transferred gene tree $T_2$, that is the tree that includes the hybrid speciation.

## 3.4 Models for reticulation events at the species level

At the species level three reticulate network models are available: *naive* ML estimation, ML with HMMs, and MP. All these methods are based on the general model of phylogenetic networks formalised by Moret *et al*. (2002).

### 3.4.1 Maximum likelihood approaches

Here we discuss two ML approaches for phylogenetic networks: *naive* ML and ML with an HMM.

#### *Naive* maximum likelihood

Likelihood in the framework of phylogeny-based reticulation detection and reconstruction was suggested for the first time by Jin and colleagues (2006). In this work, they extended the ML criterion to handle specifically HGT-oriented networks. Their extension is based on the fundamental observation that, barring reticulation, the evolutionary history of a gene is modeled by a tree, so that a phylogenetic network can be modeled by its constituent trees. This means that, the likelihood of a network $L^N(\mathbf{D}|\boldsymbol{\theta})$ is obtained as a function of the likelihoods of the trees contained in it, that is:

$$L^N(\mathbf{D}|\boldsymbol{\theta}) = \prod_{i=1}^{n} P^N(\mathbf{d}_i|\boldsymbol{\theta}) \tag{3.1}$$

where

$$P^N(\mathbf{d}_i|\boldsymbol{\theta}) = \sum_{k=1}^{K} P(T_k)\, P(\mathbf{d}_i|\boldsymbol{\theta}_k) \tag{3.2}$$

with $k = 1, \ldots, K$ indicating the number of trees within the network $N$, $P(T_k)$ denotes the probability of observing tree $T_k$, and $P(\mathbf{d}_i|\boldsymbol{\theta}_k)$, as explained in Section 2.5, is the probability of data at site $i$ for a particular tree $T_k$. The mathematical expression of $P(T_k)$ is given by

$$P(T_k) = \prod_{r \in re(T)} p_r \prod_{r \in H(N) \setminus re(T)} (1 - p_r) \tag{3.3}$$

where $p_r$ is the probability of a DNA segment being transferred along a generic edge $r$, $re(T)$ denotes the set of reticulation edges used to obtain tree $T$ in the network $N$, and $H(N)$ the set of all reticulation edges in $N$. To illustrate (3.3),

consider, once again, the example of Figure 3-3; equation (3.3) for tree $T_1$ is $P(T_1) = 1 - p_r$ as $re(T)$ is empty and $H(N) \backslash re(T)$ contains one element, and for tree $T_2$ is $P(T_2) = p_r$ as $re(T)$ contains one element and $H(N) \backslash re(T)$ is empty.

Notice that an alternative choice to (3.2) is given by the following equation

$$P^N_{\max}(\mathbf{d}_i|\boldsymbol{\theta}) = \max_{T_k \in (T_1,\ldots,T_K)} P(T_k)\, P(\mathbf{d}_i|\boldsymbol{\theta}_k)\,. \tag{3.4}$$

In this way we seek for each site that tree such that the likelihood of the leaf labels is maximised. However, the most used criterion is the summation, although the two approaches generally yield similar results. The authors consider three kind of problems: (a) the tiny problem (which means that the network topology, transition probabilities and reticulation probabilities are given), (b) the small problem (the network topology is given but not the transition and reticulation probabilities), and (c) the big problem (an initial tree is given and a set of reticulation edges is sought). Depending on the problem there is a different algorithm. For the tiny problem Jin and colleagues (2006) propose a component-wise naive algorithm. For the small version the probabilities are estimated by using hill climbing and expectation maximization (EM) algorithm. Finally for the big version a branch and bound heuristic is used.

The findings of Jin *et al*. (2006) indicate that the ML framework to model reticulation events at the species level is a promising approach, although the techniques employed are not computationally efficient and this does not allow the researcher to analyse large data sets. Also, this model does not account for dependence among sites. The next ML approach can overcome this issue by employing a more advanced model. However, if from one side it is certainly true that a more complicated model can make the method more accurate, on the other side this can impose a higher computational burden.

The ML method described in this section is implemented in NEPAL which is available in the form of executable code from `http://bioinfo.cs.rice.edu`.

**Maximum likelihood with HMMs**

ML with HMMs to model reticulation events at the species level was proposed for the first time by Snir and Tuller (2009). This new approach, called by the authors NET-HMM, captures the biologically realistic assumption that, as illustrated in Figure 2-5 of Section 2.6, neighboring sites of genomic sequences

are not independent and are more likely to belong to the same tree, which increases the accuracy of the inference. The model describes the phylogenetic network as an HMM, where each hidden state is related to one of the network's trees. Specifically, the NET-HMM is defined as a tuple $M = \{N, H\}$ where $N$ is the usual phylogenetic network, and $H$ is an HMM. The evolutionary history that is a tree in $\mathbf{T}(N)$ of every site in $\mathbf{D}$ is not known, thus a hidden state $S_i \in \{1, \ldots, K\}$ for each site in $\mathbf{D}$ is assigned. The hidden states correspond to the states of $H$. The meaning of relating the state $S_i$ of the $i^{\text{th}}$ site to a state of the HMM is that this site evolves on the tree $T_k \in \mathbf{T}(N)$ (that is the $i^{\text{th}}$ column was emitted by tree $T_k$). Let $P(S_i|S_{i-1})$ denote the transition probability between state $S_{i-1}$ and $S_i$, and $I$ be an initial state that is not related to a tree (so $S \neq I$). The likelihood of a NET-HMM model when observing a set $\mathbf{D}$ of $n$-long sequences, is defined as the probability of observing $\mathbf{D}$ which is the sum of probabilities of all length-$n$ paths of states from $\{1, \ldots, K\}$. Thus the likelihood function in (3.1) becomes

$$L^N(\mathbf{D}|\boldsymbol{\theta}, \mathbf{S}) = \sum_{S_1, S_2, \ldots, S_n} P(\mathbf{d}_1|\boldsymbol{\theta}, S_1) P(S_1|I) \prod_{i=2}^{n} P(S_i|S_{i-1}) P(\mathbf{d}_i|\boldsymbol{\theta}, S_i). \quad (3.5)$$

where $P(\mathbf{d}_i|\boldsymbol{\theta}, S_i)$ is the probability of data at site $i$ for a particular hidden state $S_i$. A different variant of (3.5) is to replace the sum by a maximum relation similarly to equation (3.4). The parameters of the NET-HMM are inferred by an algorithm which combines hill climbing in conjunction with EM.

Comparing this method to ML, the authors show in a simulation study that the NET-HMM, which accounts for dependencies among sites, performs significantly better in terms of tree allocation than the model with independence assumption. In fact, the main advantage of using the NET-HMM is its accuracy, although the computational cost is prohibitive for large data set. Hence, future work should concentrate on developing more efficient heuristics for computational optimization. Currently, the NET-HMM algorithm is not available in any user-friendly software.

### 3.4.2 Maximum parsimony

This approach, as the other two, is based on the same definition of phylogenetic network decomposition but, differently from the other two frameworks, is not likelihood-based. However, given its popularity, we decided to devote this section to illustrate the MP method in the phylogenetic network context.

MP is one of the most commonly used criteria for phylogenetic tree inference. Roughly speaking this method is based on the assumption that evolution is parsimonious, that is, the best evolutionary trees are the ones that minimise the number of changes along the edges of the tree. This criterion has been successfully used to study the evolution of various data sets for almost 30 years, and despite a heated debate concerning its performance, it is one of the most commonly used criteria for phylogeny reconstruction. Nakhleh and colleagues (2005) extended the MP criterion to phylogenetic network by using the idea that, as previously shown, a network $N$ can be decomposed in trees. Before giving any detail of the MP method for networks, we describe MP for phylogenetic trees so that the extension to networks will be easier.

Consider two strings $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ and $\mathbf{y} = (y_1, y_2, \ldots, y_n)$ of the same length $n$. The Hamming distance $H(\mathbf{x}, \mathbf{y})$ is defined as the number of positions $j$ such that $x_j \neq y_j$. For example, given two DNA sequences, $\mathbf{x} =$ AAGGCTAC and $\mathbf{y} =$ AGGGCTAT, the Hamming distance $H(\mathbf{x}, \mathbf{y}) = 2$. Given a multiple alignment of sequences $\mathbf{D}$ and a corresponding fully-labeled tree $T$, i.e., a tree in which each node $u$ is labeled by a sequence $\mathbf{d}_u$, define the Hamming distance of an edge $e \in E(T)$, denoted by $H(e)$, to be $H(\mathbf{d}_u, \mathbf{d}_v)$, where $u$ and $v$ are the two endpoints of $e$. Notice that $\mathbf{d}_u$ and $\mathbf{d}_v$ have a different meaning from $\mathbf{d}_i, (i = 1, \ldots, n)$, as the former indicate a row vector and the latter a column vector of $\mathbf{D}$. The parsimony score of a tree $T$, indicated by $T\text{Cost}(T, \mathbf{D})$, is $\sum_{e \in E(T)} H(e)$. A maximum parsimony tree for $\mathbf{D}$ is a tree which minimises the parsimony score. This definition is illustrated in Figure 3-5 where two trees, $T_1$ and $T_2$ are considered. For this case the parsimony score for each tree is calculated and is given by $T_1\text{Cost}(T_1, \mathbf{D}) = 3$ and $T_2\text{Cost}(T_2, \mathbf{D}) = 4$. Obviously, the maximum parsimony tree is $T_1$. In general computing the parsimony score of a given phylogenetic tree can be done using the Fitch's algorithm (Fitch, 1971, and Hartigan, 1973).

A natural way to extend the tree-based parsimony score to fit a data set that evolved on a network is to define the parsimony score for each site as the minimum parsimony score of that site over all trees contained in the network. This extension was first introduced by Hein (1990, 1993) in the context of meiotic recombination and then by Nakhleh *et al.*, (2005) in the context of reticulation at the species level. The expression of the parsimony score of a network $N$ leaf labeled by a set $\mathbf{D}$ of taxa is

$$N\text{Cost}(N, \mathbf{D}) = \sum_{i=1}^{n} \min_{T_k \in (T_1, \ldots, T_K)} T_k\text{Cost}\left(T_k, \mathbf{d}_i\right) \tag{3.6}$$

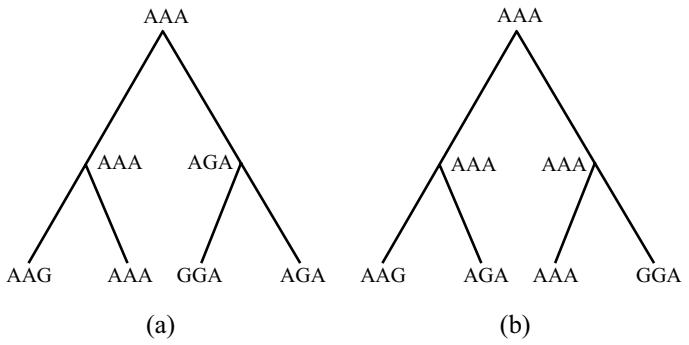



Figure 3-5: An example of MP of phylogenetic trees. The two trees in (a) and (b) are labeled by a sequence of length 3. $T_1\text{Cost}(T_1, \mathbf{D}) = 3$ and $T_2\text{Cost}(T_2, \mathbf{D}) = 4$. Based on the definition of maximum parsimony, tree $T_1$ in (a) is the optimal tree.

where Cost $(T_k, \mathbf{d}_i)$ is the parsimony score of sequence $\mathbf{d}_i$ at site $i$ on tree $T_k$. The calculation of the parsimony score of a network is illustrated in Figure 3-6 where a network $N$ of four taxa with one hybrid speciation event is decomposed in two trees, $T_1$ and $T_2$. Given the data $\mathbf{D} = (\mathbf{d}_1, \mathbf{d}_2)$, the parsimony score is equal to

$$\begin{aligned} N\text{Cost}(N, \mathbf{D}) &= \min\{T_1\text{Cost}(T_1, \mathbf{d}_1), T_2\text{Cost}(T_2, \mathbf{d}_1)\} \\ &+ \min\{T_1\text{Cost}(T_1, \mathbf{d}_2), T_2\text{Cost}(T_2, \mathbf{d}_2)\} \\ &= 1 + 1 = 2 \end{aligned}$$

This means that tree $T_1$ in Figure 3-6b is the optimal tree for site $\mathbf{d}_1$ and tree $T_2$ in Figure 3-6c is the optimal tree for site $\mathbf{d}_2$. In other words, under the MP criterion, site $\mathbf{d}_1$ evolved under tree $T_1$ and site $\mathbf{d}_2$ evolved according to tree $T_2$.

Notice that as usually large segments of DNA, rather than single sites, evolve together, expression (3.6) can be extended easily to reflect this fact, by partitioning the sequences $\mathbf{D}$ into non-overlapping blocks $b_i$ of sites, rather than sites $\mathbf{d}_i$, and replacing $\mathbf{d}_i$ by $b_i$ in it, that is

$$N\text{Cost}(N, \mathbf{D}) = \sum_{i=1}^{n} \min_{T_k \in (T_1, \ldots, T_K)} T_k\text{Cost}\left(T_k, b_i\right).$$

The algorithm used by the Nakhleh *et al.* (2005) is inapplicable to large datasets due to its demanding computational requirements. Jin and colleagues (2007a, 2007b, and 2009) devise computationally efficient solutions aimed at reconstructing and evaluating the quality of phylogenetic networks under the MP criterion. The authors show that MP currently outperforms ML, in terms

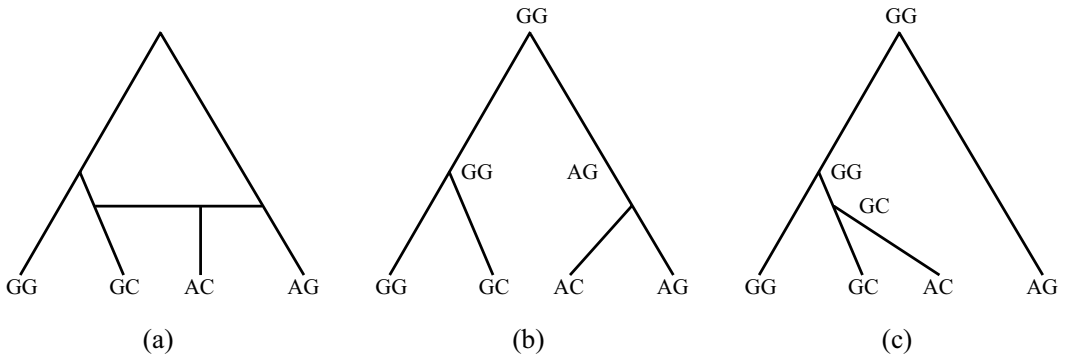



Figure 3-6: A phylogenetic network $N$ with one HS event on 4 taxa (a), each labeled by a sequence of length 2 so that there are 2 sites $\mathbf{d}_1$ and $\mathbf{d}_2$. An MP labeling of the internal nodes of the 2 trees $T_1$ in (b) and $T_2$ in (c) contained inside $N$ are shown. $T_1\text{Cost}(T_1, \mathbf{d}_1) = 1$, $T_1\text{Cost}(T_1, \mathbf{d}_2) = 2$, $T_2\text{Cost}(T_2, \mathbf{d}_1) = 2$, and $T_2\text{Cost}(T_2, \mathbf{d}_2) = 1$. Based on equation (3.6), $N\text{Cost}(N, \mathbf{D}) = \min\{T_1\text{Cost}(T_1, \mathbf{d}_1), T_2\text{Cost}(T_2, \mathbf{d}_1)\} + \min\{T_1\text{Cost}(T_1, \mathbf{d}_2), T_2\text{Cost}(T_2, \mathbf{d}_2)\} = 1 + 1 = 2$. In this case, tree $T_1$ is the optimal tree for site $\mathbf{d}_1$ and tree $T_2$ is the optimal tree for site $\mathbf{d}_2$.

of computational requirements as well as accuracy of the inferred reticulation events. However, it is important to note that the NET-HMM is as accurate as the MP approach, although the former is computationally slower than the latter. Also, since phylogenetic trees are a special case of phylogenetic networks, parsimony's shortcomings on trees are expected to be extended to phylogenetic networks. For example, long branch attraction (phenomenon that occurs when rapidly evolving lineages are inferred to be closely related, regardless of their true evolutionary relationships) could become a serious concern when employing MP. This method is implemented in NEPAL.

## 3.5 Discussion

It is well accepted and generally known that the evolutionary history of some species is not a phylogenetic tree. Rather it is a phylogenetic network, in which there have been a number of reticulate evolutionary events such as HGT and HS. Here we have reviewed three approaches for phylogenetic network reconstruction at the species level: *naive* ML, ML with HMMs, and MP. Each of these methods has strengths and weaknesses, but they can be used advantageously to combine all the information, complement one another's findings with the aim of obtaining less biased results of where reticulation events occurred. One major advantage of the first approach is its accuracy in estimating reticulation events; however its biggest problem is that it suffers from the computational

complexity of searching through the space of possible phylogenetic networks. As for the second approach, the employment of a more sophisticated model makes the method even more accurate, although the computational issue becomes more severe. The primary advantage of the last method is the computational speed, whereas the real problem is that tree estimates may have wrong branches, which result in false positive estimates of reticulation events. Yet another concern with the ML approaches and MP regards the measure of accuracy of the reticulation events. In particular MP does not give any measure of variability associated with the estimated reticulation events, whereas the two ML approaches can provide measures of accuracy via bootstrap procedures although the computational burden would increase substantially.

An alternative model which is statistically accurate and computationally fast at the same time should be developed. A Bayesian approach to phylogenetic networks seems to be a promising method for these requirements. In fact, a Bayesian method would have the advantage over the ML approaches and MP that it produces more straightforward statistical measures of phylogeny; is computationally faster, at least if compared with a maximum likelihood approach with bootstrap replicates, and the priors can take into account biological information that is otherwise inadmissible with any other approach. Hence, the next chapter will concentrate on the development of computationally efficient Markov chain Monte Carlo-based algorithms, and on the robustness of this method to model specification. This alternative approach is not directly comparable with the above methods since the input, and hence the information taken from the data is different; in our procedure all the quantities are estimated from a unique data set, whereas in the other approaches some of the parameters are estimated from previous analyses. Also, they do not infer tree topologies along a multiple DNA sequence alignment, rather they seek a set of reticulation edges.

# Chapter 4

# A Bayesian approach to phylogenetic networks

## 4.1 Introduction

Here, we present a Bayesian approach to phylogenetic networks to model reticulation events at the species level. Reticulation events in a sequence can be based on topological inference because they can lead to different topologies supported in alternative segments of the alignment. In particular, our method is based on the idea that, barring reticulation, the evolutionary history of a gene is modeled by a tree, so that a phylogenetic network can be modeled by its constituent trees (Moret *et al.*, 2004). MCMC techniques (Gilks *et al.*, 1996) are employed to estimate all the unknown quantities of the model and, allow inferences to be made regarding the number of different phylogenies for different parts of DNA sequences. Specifically, to model different phylogenies at each site two approaches are considered: naive, where the sites are treated as independent and an underlying structure of HMMs, which accounts for dependencies among adjacent sites. Also, the stochastic forward-backward algorithm, which is a single component block procedure is contrasted to the Gibbs sampler, which is a large component block procedure.

## 4.2 General method set up

Consider the following posterior probability distribution

$$P(\boldsymbol{\theta}|\mathbf{D}) = P(\mathbf{t}, \boldsymbol{\pi}, \mathbf{r}, \mathbf{S}|\mathbf{D}) = \frac{P(\mathbf{t}, \boldsymbol{\pi}, \mathbf{r}, \mathbf{S}) \times P(\mathbf{D}|\mathbf{t}, \boldsymbol{\pi}, \mathbf{r}, \mathbf{S})}{P(\mathbf{D})}, \tag{4.1}$$

where $P(\mathbf{t}, \boldsymbol{\pi}, \mathbf{r}, \mathbf{S})$ is the joint prior distribution of parameters, $P(\mathbf{D})$ is a normalising constant, and $P(\mathbf{D}|\mathbf{t}, \boldsymbol{\pi}, \mathbf{r}, \mathbf{S})$ the likelihood function (2.5) given by

$$P(\mathbf{D}|\mathbf{t}, \boldsymbol{\pi}, \mathbf{r}, \mathbf{S}) = L(\mathbf{D}|\mathbf{t}, \boldsymbol{\pi}, \mathbf{r}, \mathbf{S}). \tag{4.2}$$

$\mathbf{S} = (S_1, \ldots, S_i, \ldots, S_n)$ indicates a sequence of topologies, where $S_i \in \{1, \ldots, K\}$ represents the tree topology at site $i$ induced by $N$; $\mathbf{t}$ is the vector of the branch lengths, and $\boldsymbol{\pi}$ and $\mathbf{r}$ the parameters of the nucleotide substitution model. In other words, $S_i$ is a multinomial random variable indicating the tree topology on which the nucleotide configuration enhances. Recall that phylogenetic network $N$ can be decompsed into trees. As an illustration consider once again the example of Figure 2-2. In this case $R = 1$ (one reticulation event) and so there are $K = 2$ trees induced by the network (see Figure 2-3). Hence, $S_i$ can assume values 1 or 2 at each site. Each of these values identifies a particular tree. Indeed $S_i = 1$ refers to the underlying species tree $T_1$ that models the evolution of all genetic material that is vertically inherited from the ancestral organism and $S_i = 2$ identifies tree $T_2$ that models the evolution of horizontally transferred genetic material. Notice that the proposed method allows us to estimate a sequence of tree topologies (and other model parameters), rather than constructing a network. Estimating a network can be really challenging as different networks may display the same trees (Willson, 2010).

It is worth briefly discussing two relevant differences between the likelihood function in (2.5) and that in (4.2). First, in (4.2) the single tree topology $T$ is replaced by a sequence of topologies $\mathbf{S}$ to account for the presence of reticulation events. The second difference is in the edge lengths $\mathbf{t}$. In (2.5) $\mathbf{t}$ contains the branch lengths of $T$, whereas in (4.2), $\mathbf{t}$ can be thought of as a vector containing only the independent branch lengths of the trees rather than all the edge lengths of the trees within $N$. In fact in our example $\mathbf{t}$ has eight edge lengths not twelve.

Computing (4.1) is generally mathematically intractable. MCMC techniques can be used to approximate this probability distribution and hence to estimate the parameters of the model.

## 4.3 Prior probabilities

Inherent to the Bayesian framework is the choice of prior probabilities for all the parameters of interest ($\mathbf{t}$, $\boldsymbol{\pi}$, $\mathbf{r}$ and $\mathbf{S}$). For these parameters we make the assumption of parameter independence (Husmeier and McGuire, 2002):

$P(\mathbf{t}, \boldsymbol{\pi}, \mathbf{r}, \mathbf{S}) = P(\mathbf{t})P(\boldsymbol{\pi})P(\mathbf{r})P(\mathbf{S})$. There is no biological reason to expect the parameters to be independent, so independence is a mathematical convenience.

### 4.3.1 Branch lengths

The branch lengths $\mathbf{t}$ are defined in the usual way (Yang, 2006), that is, they represent the average number of nucleotide substitutions per site. *A priori*, they are assumed to be exponentially distributed with mean $1/\lambda$:

$$P(t|\lambda) \propto \exp(-\lambda t),\ \lambda > 0.$$

The choice of this prior is well justified in the literature and seems to work well for most analyses (e.g. Ronquist *et al.*, 2005).

### 4.3.2 Nucleotide substitution model parameters

The priors on $\boldsymbol{\pi}$ and $\mathbf{r}$ depend on the model of nucleotide substitution. In the present study, the GTR model which has eight free parameters is adopted: the nucleotide frequencies $\pi_A, \pi_C, \pi_G$ and $\pi_T$ (three free parameters because of the constraint $\sum_{x\in\Sigma} \pi_x = 1$), and relative substitution rates $(r_{xy}; x, y \in \Sigma, x \neq y)$ (five free parameters because of the constraint $\sum_{xy\in\Sigma, x\neq y} r_{xy} = 1$). In this approach nucleotide frequencies and substitution rates are assumed to be the same for all trees. This is quite plausible since the trees are induced by the same network. For both the nucleotide frequencies and the relative substitution rates a Dirichlet prior distribution is chosen. Notice that the Dirichlet parametrisation for the substitution rates is appropriate because here the substitution rates are given as a proportion of the rate sum, rather than to be scaled to the $r_{GT}$ rate.

### 4.3.3 Tree topologies

For the prior probability for a sequence of topologies $\mathbf{S} = (S_1, \ldots, S_i, \ldots, S_n)$ two alternatives are considered: naive approach and HMM structure.

#### Naive approach

We model the sites independently assuming a uniform prior on the sequence of topologies:

$$P(S_i) = 1/K,\ \forall i = 1, \ldots, n. \tag{4.3}$$

**HMM structure**

To capture dependencies between neighboring sites of the sequence we adopt a first order spatial correlation via an HMM as used by Husmeier and McGuire (2002)

$$P(\mathbf{S}) = \prod_{i=2}^{n} P(S_i|S_{i-1}, \nu)P(S_1), \tag{4.4}$$

with

$$P(S_i|S_{i-1}, \nu) = \begin{cases} \nu & \text{if} \quad S_i = S_{i-1} \\ \frac{1-\nu}{K-1} & \text{if} \quad S_i \neq S_{i-1} \end{cases}, \tag{4.5}$$

where $\nu \in (0, 1)$ is the probability of not changing topology between sites. For the initial state, $S_1$, a uniform distribution, $P(S_1) = 1/K$, is chosen. The parameter $\nu$ is a binomial random variable, for which the conjugate prior is a beta distribution with hyperparameters $\alpha$ and $\beta$

$$P(\nu|\alpha, \beta) \propto \nu^{\alpha-1}(1-\nu)^{\beta-1}, \ \alpha, \beta > 0. \tag{4.6}$$

## 4.4 Markov chain Monte Carlo sampling

If sites are modelled independently the joint distribution of the DNA sequence alignment and model parameters is given by

$$\begin{aligned} P(\mathbf{D}, \mathbf{t}, \boldsymbol{\pi}, \mathbf{r}, \mathbf{S}) = &\prod_{i=1}^{n} P(\mathbf{d}_i|\mathbf{t}, \boldsymbol{\pi}, \mathbf{r}, S_i) \\ &\times P(\mathbf{t})P(\boldsymbol{\pi})P(\mathbf{r})P(S_i) \end{aligned} \tag{4.7}$$

where $P(S_i)$ is equal to (4.3). $P(\mathbf{t})$, $P(\boldsymbol{\pi})$, and $P(\mathbf{r})$, are the prior probabilities discussed above and $P(\mathbf{d}_i|\mathbf{t}, \boldsymbol{\pi}, \mathbf{r}, S_i)$ is the probability of the $i^{th}$ column of nucleotides in the alignment, which is computed using the pruning algorithm discussed in Section 2.5. A convenient way to understand the model's dependence structure in (4.7) is via the DAGs given in Figure 4-1.

The aim is to obtain estimates of the parameters of interest. So the idea is to sample from the joint posterior distribution

$$P(\mathbf{t}, \boldsymbol{\pi}, \mathbf{r}, \mathbf{S}|\mathbf{D}). \tag{4.8}$$

To sample from (4.8), a Gibbs sampling procedure is adopted (see, e.g., Casella and George, 1992). Specifically if the superscript $(j)$ denotes the $j^{th}$ sample of

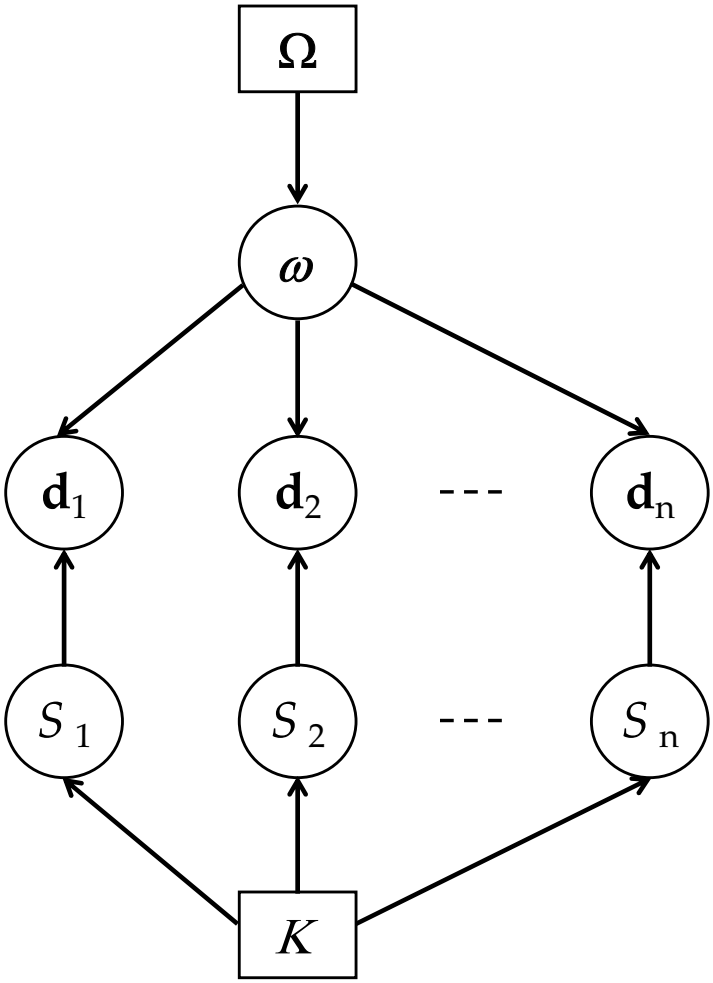



Figure 4-1: DAG representing the dependence structure of (4.7) with a uniform prior for **S**. The $\mathbf{d}_i$ represent the columns in the DNA sequence alignment, $\boldsymbol{\omega} = (\mathbf{t}, \boldsymbol{\pi}, \mathbf{r})$ where $(\mathbf{t}, \boldsymbol{\pi}, \mathbf{r})$ are described in the text, $\boldsymbol{\Omega}$ is the parameter vector that defines the prior distributions of $\mathbf{t}$, $\boldsymbol{\pi}$ and $\mathbf{r}$; the $S_i$ represent the tree topologies, and $K$ is the parameter defining the prior distribution of $S_i$.

the Markov chain, the $(j+1)^{th}$ sample is obtained as follows:

$$\begin{aligned}
\mathbf{t}^{(j+1)} &\sim P(\cdot|\boldsymbol{\pi}^{(j)}, \mathbf{r}^{(j)}, \mathbf{S}^{(j)}, \mathbf{D}) \\
\boldsymbol{\pi}^{(j+1)} &\sim P(\cdot|\mathbf{t}^{(j+1)}, \mathbf{r}^{(j)}, \mathbf{S}^{(j)}, \mathbf{D}) \\
\mathbf{r}^{(j+1)} &\sim P(\cdot|\mathbf{t}^{(j+1)}, \boldsymbol{\pi}^{(j+1)}, \mathbf{S}^{(j)}, \mathbf{D}) \\
\mathbf{S}^{(j+1)} &\sim P(\cdot|\mathbf{t}^{(j+1)}, \boldsymbol{\pi}^{(j+1)}, \mathbf{r}^{(j+1)}, \mathbf{D}).
\end{aligned} \tag{4.9}$$

The order of these steps is arbitrary but with the superscripts changed accordingly.

If the dependency between neighboring sites is modelled, then the joint distribution of the data and the parameters is given by

$$\begin{aligned}
P(\mathbf{D}, \mathbf{t}, \boldsymbol{\pi}, \mathbf{r}, \mathbf{S}, \nu) = &\prod_{i=1}^{n} P(\mathbf{d}_i|\mathbf{t}, \boldsymbol{\pi}, \mathbf{r}, S_i) \\
&\times P(\mathbf{t})P(\boldsymbol{\pi})P(\mathbf{r})P(\mathbf{S})P(\nu).
\end{aligned} \tag{4.10}$$

The structure is similar to that of the naive approach, but with $P(\mathbf{S})$ equal to (4.4), and the additional prior $P(\nu)$ given by (4.6). Figure 4-2 represents the model's dependence structure in (4.10).

Because of the dependence structure between adjacent sites the joint posterior distribution is given by

$$P(\mathbf{t}, \boldsymbol{\pi}, \mathbf{r}, \mathbf{S}, \nu|\mathbf{D}). \tag{4.11}$$

Similarly to (4.9), sampling from (4.11) is obtained as follows:

$$\begin{aligned}
\mathbf{t}^{(j+1)} &\sim P(\cdot|\boldsymbol{\pi}^{(j)}, \mathbf{r}^{(j)}, \mathbf{S}^{(j)}, \nu^{(j)}, \mathbf{D}) \\
\boldsymbol{\pi}^{(j+1)} &\sim P(\cdot|\mathbf{t}^{(j+1)}, \mathbf{r}^{(j)}, \mathbf{S}^{(j)}, \nu^{(j)}, \mathbf{D}) \\
\mathbf{r}^{(j+1)} &\sim P(\cdot|\mathbf{t}^{(j+1)}, \boldsymbol{\pi}^{(j+1)}, \mathbf{S}^{(j)}, \nu^{(j)}, \mathbf{D}) \\
\mathbf{S}^{(j+1)} &\sim P(\cdot|\mathbf{t}^{(j+1)}, \boldsymbol{\pi}^{(j+1)}, \mathbf{r}^{(j+1)}, \nu^{(j)}, \mathbf{D}) \\
\nu^{(j+1)} &\sim P(\cdot|\mathbf{t}^{(j+1)}, \boldsymbol{\pi}^{(j+1)}, \mathbf{r}^{(j+1)}, \mathbf{S}^{(j+1)}, \mathbf{D}).
\end{aligned} \tag{4.12}$$

### 4.4.1 Branch lengths and nucleotide substitution model parameters

For sampling the parameters $\mathbf{t}$, $\boldsymbol{\pi}$ and $\mathbf{r}$ the Metropolis-Hastings algorithm is applied (Chib and Greenberg, 1995). Let $\boldsymbol{\theta}^{(j)}$ denote the parameter configuration in the $j^{th}$ sampling step. A new parameter configuration $\boldsymbol{\theta}^{*}$ is sampled

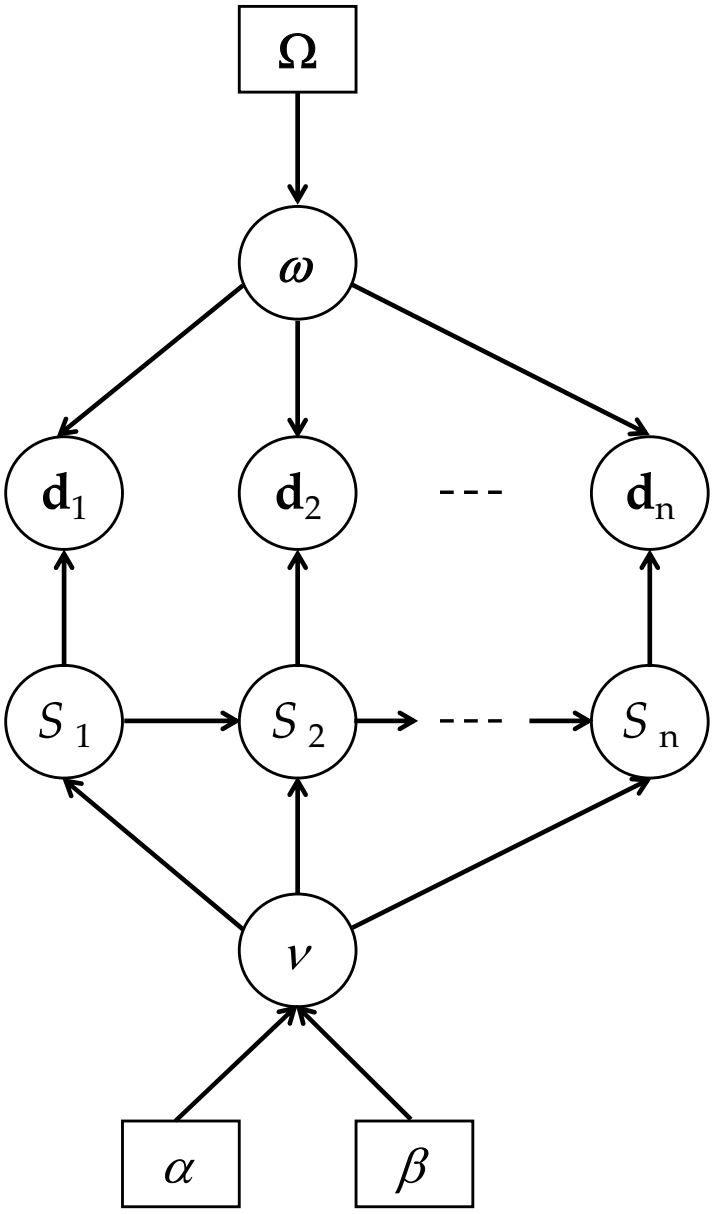



Figure 4-2: DAG representing the dependence structure of (4.10) with an HMM for **S**. The $\mathbf{d}_i$ represent the columns in the DNA sequence alignment, $\boldsymbol{\omega} = (\mathbf{t}, \boldsymbol{\pi}, \mathbf{r})$ where $(\mathbf{t}, \boldsymbol{\pi}, \mathbf{r})$ are described in the text, $\boldsymbol{\Omega}$ is the parameter vector that defines the prior distributions of $\mathbf{t}, \boldsymbol{\pi}$ and $\mathbf{r}$; the $S_i$ represent the tree topologies, $\nu$ is a parameter that defines the priori distributions of the $S_i$, and $\alpha$ and $\beta$ are hyperparameters defining the prior distribution of $\nu$.

from a proposal distribution $Q(\boldsymbol{\theta}^*|\boldsymbol{\theta}^{(j)})$, and then accepted with probability:

$$a(\boldsymbol{\theta}^*) = \min\left\{\frac{P(\boldsymbol{\theta}^*)Q(\boldsymbol{\theta}^{(j)}|\boldsymbol{\theta}^*)}{P(\boldsymbol{\theta}^{(j)})Q(\boldsymbol{\theta}^*|\boldsymbol{\theta}^{(j)})}, 1\right\}, \tag{4.13}$$

in which case $\boldsymbol{\theta}^{(j+1)} = \boldsymbol{\theta}^*$, otherwise $\boldsymbol{\theta}^{(j+1)} = \boldsymbol{\theta}^{(j)}$. The distribution $P$ is given by (4.8) or (4.11), depending on the prior choice of $\mathbf{S}$. In theory the algorithm converges to the posterior distribution irrespective of the choice of the proposal distribution (Gilks *et al.*, 1996). However, in practice the choice of the proposal distribution is crucial to achieve convergence within a reasonable number of iterations.

For the components $t$ of the vector of branch lengths $\mathbf{t}$, a new value is selected with a proportional shrinking and expanding method (Yang, 2006). This means that the proposed branch length $t^*$ is given by $t^* = t^{(j)}c$, where $c = \exp\{\epsilon(U - 0.5)\}$ and $U$ is uniformly distributed on $[0, 1]$, with $\epsilon > 0$ to be a small tuning parameter. The proposal ratio $Q(\boldsymbol{\theta}^{(j)}|\boldsymbol{\theta}^*)/Q(\boldsymbol{\theta}^*|\boldsymbol{\theta}^{(j)})$ in (4.13) is $c$. To see this, derive $Q(t^*|t^{(j)})$ through variable transform, considering $t^*$ as a function of $U$ while treating $\epsilon$ and $t^{(j)}$ as fixed. Since $U = 1/2 + \log(t^*/t^{(j)})/\epsilon$, and $\mathrm{d}U/\mathrm{d}t_l^* = 1/(\epsilon t^*)$, we have

$$Q\left(t^*|t^{(j)}\right) = f\left\{U\left(t^*\right)\right\} \times \left|\frac{\mathrm{d}U}{\mathrm{d}t^*}\right| = \frac{1}{\epsilon\left|t^*\right|}.$$

Similarly $Q(t^{(j)}|t^*) = 1/(\epsilon|t^{(j)}|)$, so the proposal ratio is

$$\frac{Q\left(t^{(j)}|t^*\right)}{Q\left(t^*|t^{(j)}\right)} = c.$$

For the nucleotides frequencies and rates of substitution, new values are sampled from a Dirichlet distribution. This ensures that the normalisation constraints, $\sum_{x\in\Sigma}\pi_x = 1$ and $\sum_{x,y\in\Sigma,x\neq y} r_{xy} = 1$, are satisfied. The parameters of the Dirichlet distribution are chosen proportional to the current values of the nucleotides frequencies and the rates of substitution. In particular, if the current values are $z_1^{(j)}, \ldots, z_k^{(j)}$ where $\sum_i z_i^{(j)} = c$, we let $z^* = cY$ where $Y$ is randomly chosen from a Dirichlet distribution with parameters $(\delta z_1^{(j)}, \delta z_2^{(j)}, \ldots, \delta z_k^{(j)})$, with $\delta$ to be a tuning parameter (Larget and Simon, 1999). The proposal ratio is the ratio of two Dirichlet densities:

$$\frac{Q\left(z^{(j)}|z^*\right)}{Q\left(z^*|z^{(j)}\right)} = \prod_{i=1}^{k}\frac{\Gamma(\delta z_i^{(j)})z_i^{\delta z_i^*-1}}{\Gamma\left(\delta z_i^*\right)z_i^{*\delta z_i^{(j)}-1}}.$$

Note that, because of the existence of computational trapping states, there might be some problems with this updating mechanism. As explained by Loza-Reyes (2010), a drawback with this proposal is that as $z_i \to 0$ the chain may fall into a trap near the zero-boundary (that is, the proposal will generate very small steps, all within the neighbourhood of 0) as $\mathbb{E}(z_i) \to 0$ and $Var(z_i) \to 0$. To overcome this issue, Loza-Reyes (2010) shifts the centre of the Dirichlet proposal by a small quantity $\epsilon > 0$. The $\epsilon$-corrected algorithm can hence escape from trapping states at the zero-boundary, without resorting to sophisticated tempered schemes which create extra computational burden.

### 4.4.2 Tree topologies

For sampling the state sequences **S** two approaches can be adopted:

1. the Gibbs sampling algorithm;

2. the stochastic forward-backward algorithm.

#### Gibbs sampling algorithm

Within the Gibbs sampling scheme each state $S_i$ can be sampled separately conditional on the others:

$$\begin{array}{rcl} S_1^{(j+1)} & \sim & P(\cdot|S_2^{(j)}, S_3^{(j)}, \ldots, S_n^{(j)}, \mathbf{t}^{(j+1)}, \boldsymbol{\pi}^{(j+1)}, \mathbf{r}^{(j+1)}, \nu^{(j)}, \mathbf{D}) \\ S_2^{(j+1)} & \sim & P(\cdot|S_1^{(j+1)}, S_3^{(j)}, \ldots, S_n^{(j)}, \mathbf{t}^{(j+1)}, \boldsymbol{\pi}^{(j+1)}, \mathbf{r}^{(j+1)}, \nu^{(j)}, \mathbf{D}) \\ & \ldots & \\ S_n^{(j+1)} & \sim & P(\cdot|S_1^{(j+1)}, S_2^{(j+1)}, \ldots, S_{n-1}^{(j+1)}, \mathbf{t}^{(j+1)}, \boldsymbol{\pi}^{(j+1)}, \mathbf{r}^{(j+1)}, \nu^{(j)}, \mathbf{D}). \end{array} \tag{4.14}$$

The computational complexity of (4.14) is reduced considerably by assumption (4.3) which implies that

$$\begin{aligned} &P(S_i|S_1, \ldots, S_{i-1}, S_{i+1}, \ldots, S_n, \mathbf{t}, \boldsymbol{\pi}, \mathbf{r}, \nu, \mathbf{D}) = \\ &P(S_i|\mathbf{t}, \boldsymbol{\pi}, \mathbf{r}, \mathbf{d}_i) \propto P(\mathbf{d}_i|\mathbf{t}, \boldsymbol{\pi}, \mathbf{r}, S_i). \end{aligned} \tag{4.15}$$

Note that (4.15) can be easily normalised to give a proper probability, from which sampling is straightforward.

By considering the first order spatial correlation structure for the sequence of the topologies, (4.14) looks a little more complicated but is still relatively

simple. In fact:

$$\begin{aligned} &P(S_i|S_1,\ldots,S_{i-1},S_{i+1},\ldots,S_n,\mathbf{t},\boldsymbol{\pi},\mathbf{r},\nu,\mathbf{D}) = \\ &P(S_i|S_{i-1},S_{i+1},\mathbf{t},\boldsymbol{\pi},\mathbf{r},\nu,\mathbf{d}_i) \propto \\ &P(S_{i+1}|S_i,\nu)P(S_i|S_{i-1},\nu)P(\mathbf{d}_i|\mathbf{t},\boldsymbol{\pi},\mathbf{r},S_i), \end{aligned} \tag{4.16}$$

where $P(S_{i+1}|S_i,\nu)$ and $P(S_i|S_{i-1},\nu)$ are given by (4.5). Again, (4.16) can be easily normalised to give a proper probability, from which sampling is easy. However the Gibbs sampling algorithm with HMM might encounter slow convergence and poor mixing problems due to the large number of component blocks. The former means that it takes a very long time for the chain to reach stationary. The latter means that the sampled states are highly correlated over iterations and the chain is inefficient in exploring the parameter space.

**Stochastic forward-backward algorithm**

The stochastic forward-backward algorithm is an alternative simulation strategy which makes use of a different conditional independence property of the model (Boys *et al*., 2000). It has the advantage of being a single component block, i.e. simulating a full sequence each time, which helps the convergence of the sampler.

Let us define

$$\alpha_m = P(\mathbf{d}_1,\ldots,\mathbf{d}_m,S_m) \tag{4.17}$$

where the dependence on $\mathbf{t}$, $\boldsymbol{\pi}$, $\mathbf{r}$ and $\nu$ is dropped in order to simplify the notation. (4.17) can be written as

$$\begin{aligned} \alpha_m &= P(\mathbf{d}_1,\ldots,\mathbf{d}_m,S_m) \\ &= \sum_{S_1}\cdots\sum_{S_{m-1}} P(\mathbf{d}_1,\ldots,\mathbf{d}_m,S_1,\ldots,S_{m-1},S_m) \\ &= \sum_{S_1}\cdots\sum_{S_{m-1}}\prod_{i=1}^{m} P(\mathbf{d}_i|S_i)P(S_i|S_{i-1}) \\ &= \sum_{S_1}\cdots\sum_{S_{m-1}} P(\mathbf{d}_m|S_m)P(S_m|S_{m-1})\prod_{i=1}^{m-1} P(\mathbf{d}_i|S_i)P(S_i|S_{i-1}) \\ &= P(\mathbf{d}_m|S_m)\sum_{S_{m-1}} P(S_m|S_{m-1})\sum_{S_1}\cdots\sum_{S_{m-2}}\prod_{i=1}^{m-1} P(\mathbf{d}_i|S_i)P(S_i|S_{i-1}) \\ &= P(\mathbf{d}_m|S_m)\sum_{S_{m-1}} P(S_m|S_{m-1})\alpha_{m-1}(S_{m-1}), \end{aligned} \tag{4.18}$$

which is the function computed in the forward pass of the forward-backward algorithm for HMMs. In practice, for $i = 1$, $\alpha_1$ is calculated using $P(\mathbf{d}_1|S_1)$ only, which in turn is computed via the pruning algorithm (see Section 2.5). All quantities in 4.18 are easy to calculate; the first term is the probability of the data at site $m$, the second term the HMM prior, and the last can be calculated from the previous value of $\alpha$. To calculate the function in the backward pass, we can write

$$\begin{aligned}
&P(S_i|S_{i+1},\ldots,S_n,\mathbf{d}_1,\ldots,\mathbf{d}_n)\\
&\propto P(S_i,S_{i+1},\ldots,S_n,\mathbf{d}_1,\ldots,\mathbf{d}_n)\\
&= P(\mathbf{d}_{i+1},\ldots,\mathbf{d}_n,S_{i+1},\ldots,S_n,|S_i,\mathbf{d}_i,\mathbf{d}_1,\ldots,\mathbf{d}_i)P(S_i,\mathbf{d}_1,\ldots,\mathbf{d}_i)\\
&= P(\mathbf{d}_{i+1},\ldots,\mathbf{d}_n,S_{i+1},\ldots,S_n,|S_i)\alpha_i(S_i)\\
&= P(\mathbf{d}_{i+1},\ldots,\mathbf{d}_n,S_{i+2},\ldots,S_n,|S_{i+1})P(S_{i+1}|S_i)\alpha_i(S_i)\\
&\propto P(S_{i+1}|S_i)\alpha_i(S_i).
\end{aligned} \tag{4.19}$$

The simplifications carried out in (4.19) follow directly from the independence relations in HMMs (see Figure 4-2). The last step follows from the fact that the first term in the second last line is independent of $S_i$. Hence, the function computed in the backward algorithm, derived from (4.19), can be written as

$$P(S_i = k|S_{i+1},\ldots,S_n,\mathbf{d}_1,\ldots,\mathbf{d}_n) = \frac{P(S_{i+1}|S_i = k)\alpha_i(S_i = k)}{\sum_l P(S_{i+1}|S_i = l)\alpha_i(S_i = l)}. \tag{4.20}$$

Obviously, any scaling constant also cancels out in the normalisation; hence replacing $\alpha_i(S_i)$ by some scaled version for numerical stabilisation of the forward algorithm will not affect the result. The backward algorithm is initialised by drawing the initial state, $S_n$, from the following distribution:

$$P(S_n = k|\mathbf{d}_1,\ldots,\mathbf{d}_n) = \frac{\alpha_n(S_n = k)}{\sum_l \alpha_n(S_n = l)}. \tag{4.21}$$

The overall algorithm can thus be summarised as follows:

1. For $i = 1$, $\alpha_1 = P(\mathbf{d}_1|S_1)$.
2. For $i = 2,\ldots,n$ run the (scaled) forward algorithm (4.18).
3. Sample $S_n$ from (4.21).
4. Sample the remaining states $S_{n-1}, S_{n-2},\ldots,S_2,S_1$ recursively from (4.20).

Below is a sketch of the structure of the stochastic forward-backward algorithm which has been implemented in R:

```
n <- sequence length
alpha[1] <- probability of observing the data at site 1

for(i in 2:n){
 alpha[i] <- equation (4.18)
             }

ProbS[n] <- equation (4.21)
Sample S_n from ProbS[n]

for(i in (n-1):1){
 ProbS[i] <- equation (4.20)
 Sample S_n-1, S_n-2,..., S_1  from ProbS[n-1], ProbS[n-2],..., ProbS[1]
                 }
```

Notice that the the algorithm described here is not applicable to MRFs because it requires the prior for the tree topologies to depend on one neighbor only, $P(S_i|S_{i-1})$, whereas an MRF prior models dependencies between sites accounting for both neighbors, $P(S_i|S_{i-1}, S_{i+1})$.

### 4.4.3 Probability of not changing topology $\nu$

Let us define $\Psi = \sum_{i=2}^{n} \delta(S_{i-1}, S_i)$, where $\delta(S_{i-1}, S_i)$ denotes the Kronecker delta function, which is 1 when $S_{i-1} = S_i$ and 0 otherwise, from (4.5) and (4.6) it is easy to show that writing the joint probability distribution in function of $\nu$ gives $P(\mathbf{D}, \mathbf{t}, \boldsymbol{\pi}, \mathbf{r}, \mathbf{S}, \nu) \propto \nu^{\Psi+\alpha-1}(1-\nu)^{n-\Psi+\beta-2}$. Upon normalisation this gives a beta distribution

$$P(\nu|\mathbf{t}, \boldsymbol{\pi}, \mathbf{r}, \mathbf{S}, \mathbf{D}) \propto \nu^{\Psi+\alpha-1}(1-\nu)^{n-\Psi+\beta-2} \tag{4.22}$$

from which sampling is straightforward.

## 4.5 Discussion

In this chapter a Bayesian approach to phylogenetic networks based on the definition of network by Moret *et al*. (2004) has been presented. The idea behind this approach is to introduce a (hidden) state representing the tree topologies induced by the network at a given site. A state transition from one topology into another corresponds to a single or multiple reticulation events. To

model the tree topologies at each site two approaches have been considered: (1) naive, where the sites are modelled independently, and (2) a first order hidden Markov chain to account for the dependency structure of adjacent sites. MCMC techniques are employed to compute all posterior quantities of interest and allow inferences to be made regarding the number of topology types along a multiple DNA sequence alignment. Specifically, two procedures for sampling the state sequences $\mathbf{S}$ have been contrasted: the Gibbs sampler and the forward-backward algorithm. To compute the posterior quantities of the remaining parameters Metropolis-Hastings algorithms have been proposed. Notice that our proposed method is related to the work by Song and Hein (2005) in that both methods recover the actual site-specific evolutionary relationships. However Song and Hein (2005) proposed this approach in the context of ancestral recombination graphs.

The algorithm described here has been written in `R` (`www.r-project.org`). Its validity will be tested on synthetic data in the next chapter. In Chapter 6 this approach will be used for analysing a biological dataset whose evolution includes horizontal gene transfers.

# Chapter 5

# Simulation study

## 5.1 Introduction

In this chapter we investigate the empirical performance of the method described in Chapter 4 on simulated data. First, we test the algorithm on its ability to 1) correctly classify tree topologies along aligned sequences and 2) recover the true synthetic parameter values. For both points, we contrast the naive prior for the sequence of topologies to the HMM prior. Then we compare the performance of the Gibbs sampling and the forward-backward algorithm for the sequence phylogenies in terms of convergence and mixing. Finally, we present several scenarios and misspecifications with the aim of getting some insights in terms of practical implications.

## 5.2 Data generating process

DNA sequences, 600 bases long, are evolved along the network shown in Figure 5-1, using the GTR model (2.3) of nucleotide substitution with:

- nucleotide frequencies $\boldsymbol{\pi} = (0.10, 0.40, 0.10, 0.40)$;
- rates of substitution $\mathbf{r} = (0.09, 0.22, 0.12, 0.14, 0.35, 0.040)$;
- branch edges
  $\mathbf{t} = (0.09, 0.15, 0.10, 0.20, 0.10, 0.20, 0.15, 0.10, 0.25, 0.15, 0.10, 0.20, 0.30, 0.40)$.

This network contains two horizontal gene transfer events inducing four tree topologies (see Figure 5-2) in four different regions; the first region (between nucleotides $i = 1 - 150$) does not include any HGTs, the second (between nucleotides $i = 151 - 300$) contains one HGT, the third (between nu-

cleotides $i = 301 - 450$) includes another HGT, and the last (between nucleotides $i = 451 - 600$) incorporates both HGTs. The sequence alignment is generated with the program `SEQ-GEN` (Rambaut and Grassly, 1997). Specifically, we generated the DNA sequences so that the first 150 sites have been generated under tree $T_1$, the second 150 sites under tree $T_2$, the third under tree $T_3$, and the last 150 under tree $T_4$.[1]

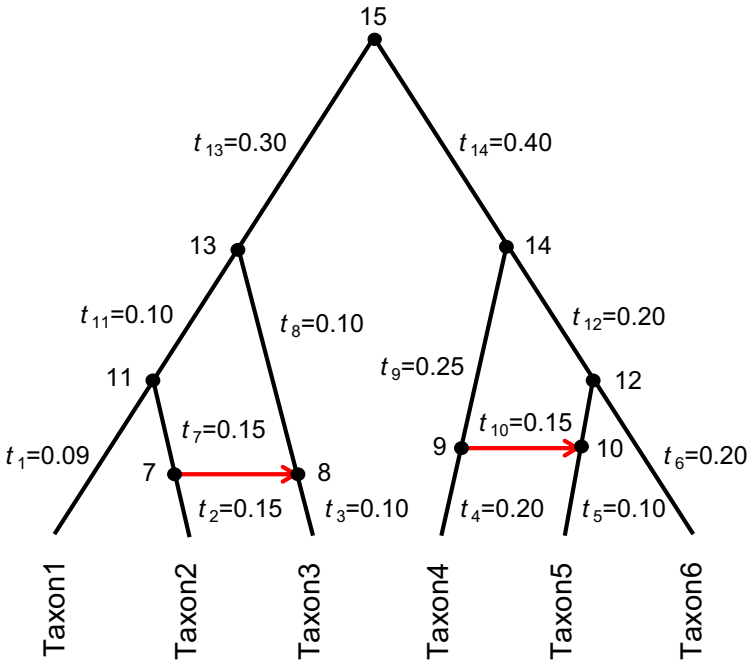


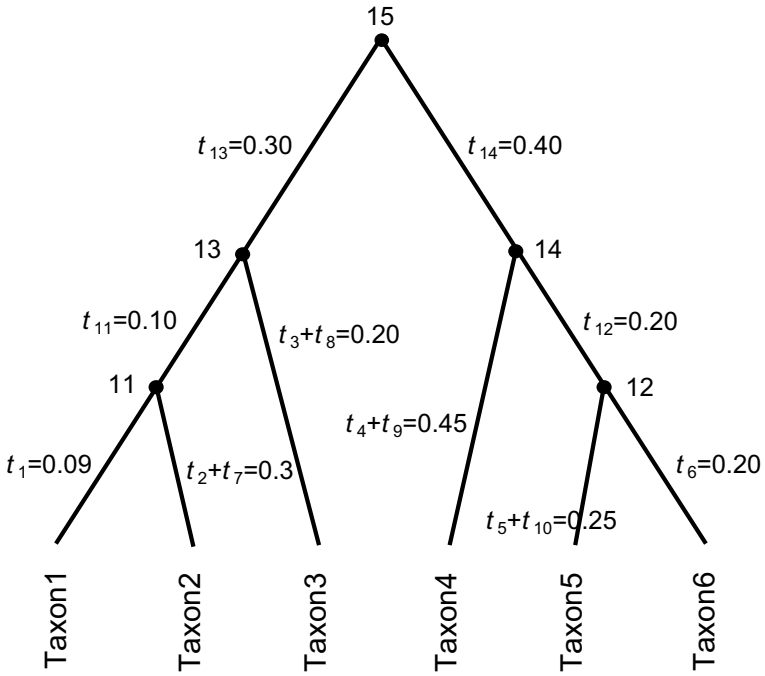

Figure 5-1: Phylogenetic network of six taxa with two reticulation edges ($R = 2$) and fourteen branch lengths ($t_1$-$t_{14}$).

### 5.2.1 Choice of prior parameters

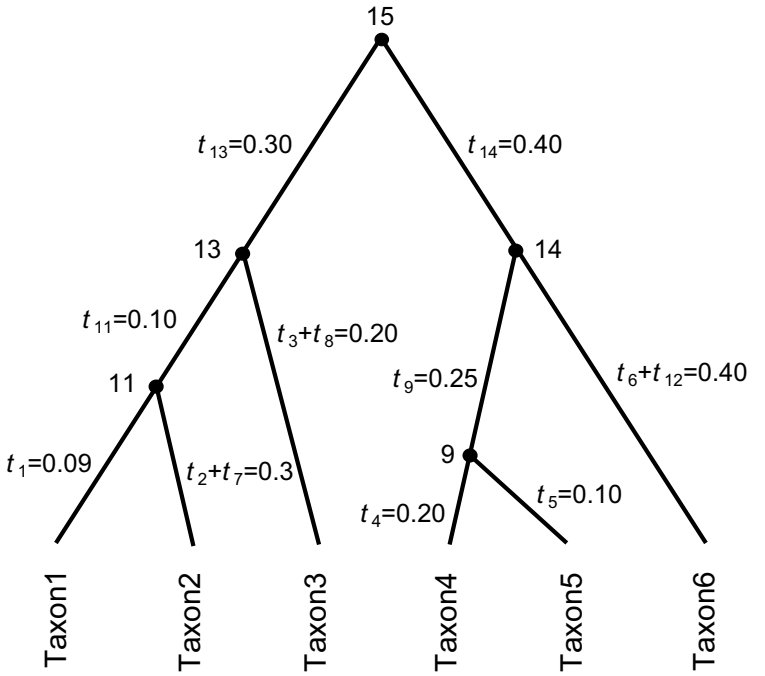
#### Branch lengths

*A priori* the branch edges are assumed to be exponentially distributed with parameter $\lambda = 10$, a choice which seems to work well in practice (Ronquist *et al.*, 2005)

[1] The `SEQ-GEN` code lines used to generate the data are:
seq-gen -mREV -f 0.10,0.40,0.10,0.40 -r0.11,0.22,0.12,0.14,0.35,0.060 -p4 -l600 <network> data
where `network` contains the four tree topologies which in Newick format (Felsenstein, 2009) are defined as:
[150](((Taxon1:0.09,Taxon2:0.30):0.1,Taxon3:0.2):0.3,(Taxon4:0.45,(Taxon5:0.25,Taxon6:0.2):0.2):0.4)
[150](((Taxon1:0.09,Taxon2:0.30):0.1,Taxon3:0.2):0.3,((Taxon4:0.2,Taxon5:0.1):0.25,Taxon6:0.4):0.4)
[150]((Taxon1:0.09,(Taxon2:0.15,Taxon3:0.1):0.15):0.4,(Taxon4:0.45,(Taxon5:0.25,Taxon6:0.2):0.2):0.4)
[150]((Taxon1:0.09,(Taxon2:0.15,Taxon3:0.1):0.15):0.4,((Taxon4:0.2,Taxon5:0.1):0.25,Taxon6:0.4):0.4).

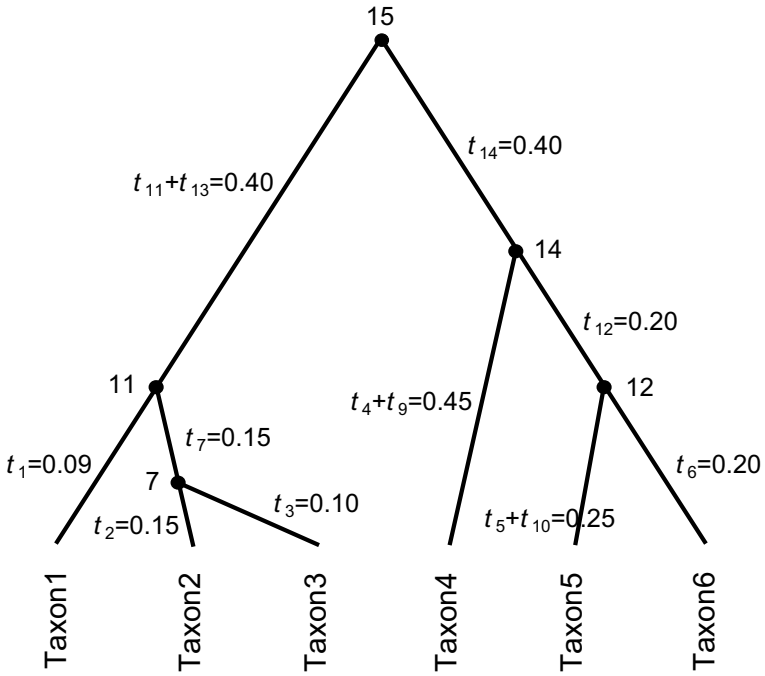



(a) The underlying species tree $T_1$, that is, the tree that does not include any reticulation edges.

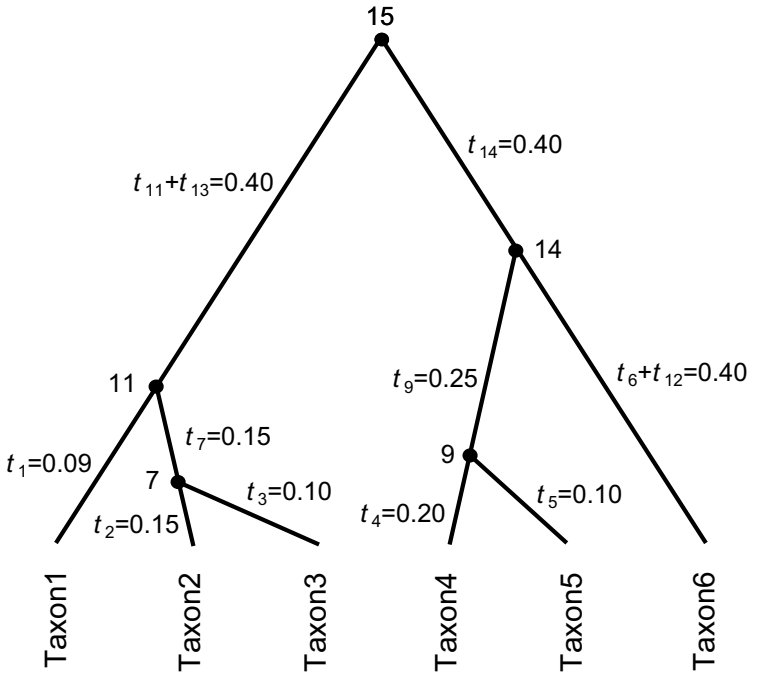



(b) The horizontally transferred gene tree $T_2$ that includes the reticulation edge $(9, 10)$.

Figure 5-2: Trees induced by the network described in Figure 5-1.




(c) The horizontally transferred gene tree $T_3$ that includes the reticulation edge $(7, 8)$.




(d) The horizontally transferred gene trees $T_4$ that includes both the reticulation edges $(7, 8)$ and $(9, 10)$.

Figure 5-2: Trees induced by the network described in Figure 5-1 (con't).

#### Nucleotide substitution model parameters

As seen in Section 4.3 a natural prior for $\boldsymbol{\pi}$ and $\mathbf{r}$ is a Dirichlet distribution. Specifically, for the nucleotide frequencies $\boldsymbol{\pi}$ we choose a Dirichlet(1,1,1,1), which is a uniform distribution subject to the normalisation constraint and thus non-informative. Similarly for the rates of substitution $\mathbf{r}$, where we choose a Dirichlet(1,1,1,1,1,1) distribution.

#### Tree topologies

As discussed in Section 4.3 for the prior probability of $\mathbf{S}$ two alternatives are possible:

1. the naive approach where (4.3) is equal to $1/4$;
2. the HMM where both $\alpha$ and $\beta$ are set to 1 allowing (4.6) to be uniform over the interval $[0, 1]$, and thus defining a non-informative prior.

### 5.2.2 Tuning parameter setting

#### Branch lengths

To select new values of the branch lengths via the proportional shrinking and expanding method described in Section 4.4, it is important that the tuning parameter $\epsilon$ is chosen to achieve a reasonable acceptance rate; too small a value of $\epsilon$ means that the proposed states will be very close to the current state, and most proposals will be accepted. However too large a value of $\epsilon$ might cause most proposals to fail in unreasonable regions of the parameter space and to be rejected. A number of runs, each with a different value for $\epsilon$ have been performed. Table 5.1 displays the ergodic averages of the branch lengths corresponding to 100000 samples after 20000 iterations of burn-in. The worst-performing samplers are those with $\epsilon = 0.01$ and $\epsilon > 0.06$. The reason for this is that, as already explained, it is important to tune $\epsilon$ so that the acceptance rate is neither to small nor too big.

#### Nucleotide substitution model parameters

In order to achieve a good acceptance ratio, the values of the tuning parameter $\delta$ for the proposal distributions of $\boldsymbol{\pi}$ and $\mathbf{r}$ are set to 300 and 1000 respectively. Other choices of parameter did not perform well. The bad estimation performance is due to the zero-stickiness issue explained by Loza-Reyes (2010).

| Edge lengths | $\epsilon = 0.01$ | $\epsilon = 0.02$ | $\epsilon = 0.04$ | $\epsilon = 0.06$ | $\epsilon = 0.07$ | $\epsilon = 0.08$ | True |
|---|---|---|---|---|---|---|---|
| $t_1$ | 0.05 | 0.07 | 0.07 | 0.06 | 0.03 | 0.01 | 0.09 |
| $t_2$ | 0.16 | 0.14 | 0.13 | 0.14 | 0.10 | 0.10 | 0.15 |
| $t_3$ | 0.07 | 0.09 | 0.09 | 0.10 | 0.09 | 0.09 | 0.10 |
| $t_4$ | 0.21 | 0.21 | 0.20 | 0.20 | 0.18 | 0.22 | 0.20 |
| $t_5$ | 0.13 | 0.09 | 0.08 | 0.09 | 0.07 | 0.05 | 0.10 |
| $t_6$ | 0.23 | 0.19 | 0.19 | 0.19 | 0.18 | 0.25 | 0.20 |
| $t_7$ | 0.12 | 0.17 | 0.17 | 0.17 | 0.17 | 0.17 | 0.15 |
| $\tilde{t}_8$ | 0.24 | 0.20 | 0.21 | 0.20 | 0.22 | 0.26 | 0.20 |
| $t_9$ | 0.20 | 0.27 | 0.28 | 0.26 | 0.29 | 0.30 | 0.25 |
| $\tilde{t}_{10}$ | 0.29 | 0.26 | 0.27 | 0.26 | 0.30 | 0.20 | 0.25 |
| $t_{11}$ | 0.05 | 0.07 | 0.08 | 0.07 | 0.03 | 0.04 | 0.10 |
| $t_{12}$ | 0.22 | 0.23 | 0.23 | 0.22 | 0.23 | 0.15 | 0.20 |
| $t_{13}$ | 0.35 | 0.33 | 0.34 | 0.33 | 0.24 | 0.38 | 0.30 |
| $t_{14}$ | 0.32 | 0.37 | 0.36 | 0.37 | 0.45 | 0.47 | 0.40 |

Table 5.1: Ergodic averages for the branch lengths for 100000 samples after a burn-in. These values are reported for 6 runs, each run with a different $\epsilon$ value.

## 5.3 Simulation results using naive approach

### 5.3.1 Inferring tree topologies

Algorithm (4.9) was run for 600000 iterations with the first 100000 discarded as burn-in.

In this subsection we test the performance of the naive classifier that assigns the uniform prior on the sequence topologies (4.3). The first four plots from the top in Figure 5-3 show the posterior probabilities for the four topologies against the site $i$ in the multiple alignment where the four regions, discussed above, are framed by vertical dashed lines. The bottom row of Figure 5-3 shows the barplots obtained from the mean of the posterior probability of **S** for the four topologies which summarise the information contained in the top plots and can be used for classification purposes. Looking at this figure, the probabilities exhibit very noisy patterns, and hence the classification performance is rather disappointing. This is the result of the poor prior, $P(S_i) = 1/4$, which does not account for correlations between adjacent sites.

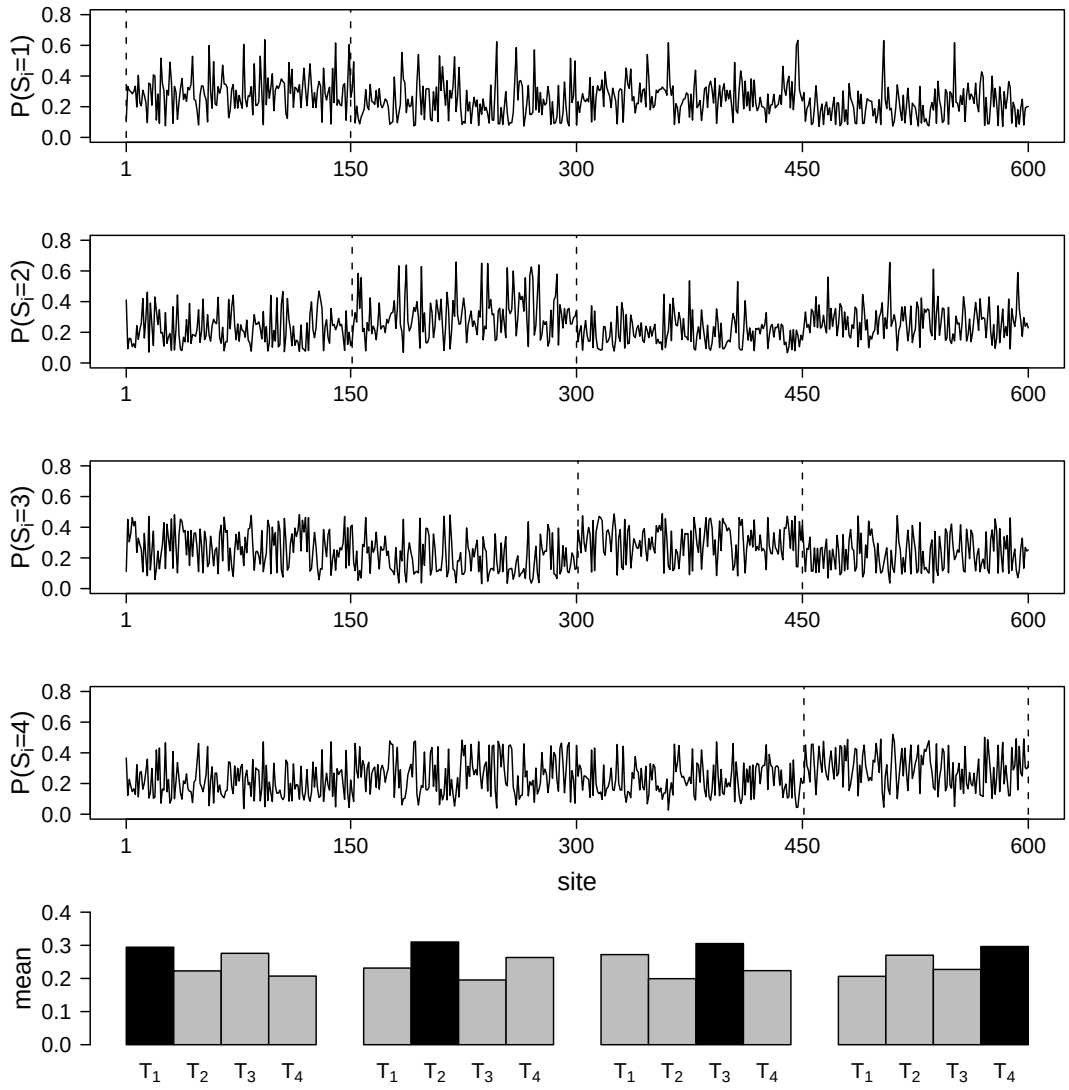



Figure 5-3: Inferred HGTs with naive approach. The first four plots from the top show the posterior probabilities for the four tree topologies (indicated for simplicity by $P(S_i = k),\ k = 1, 2, 3, 4$). The bar plots (bottom row) show the mean of the posterior probabilities of **S** for the four topologies in each region.

### 5.3.2 Parameter estimates

Table 5.2 reports the estimates for the fourteen branch lengths. The fact that $\tilde{t}_8 = t_3 + t_8$ and $\tilde{t}_{10} = t_5 + t_{10}$ implies that $\tilde{t}_8 > t_3$ and $\tilde{t}_{10} > t_5$ (Snir and Tuller, 2009). From the point estimates we can see that this is the case, although a constrained sampling scheme would be more appropriate to ensure that these conditions are exactly met. In the context of phylogenetic networks, Jin *et al.* (2006) stressed that the use of simpler models as opposed to more complicated realistic ones can reduce considerably the running time and complexity of the algorithm, while still providing reasonable important practical results. In the same way, we did not pursue a more complicated approach whose feasibility can be explored in future research.

From the results we can see that the algorithm does not recover well the true synthetic parameter values; the estimates are not always close to the true values and the credible intervals are wide. This comes as no surprise as this algorithm does not properly allocate sites to trees. Different conclusions can be drawn for the estimates of the nucleotide frequencies and rates of substitution. In fact these estimates are close to the true vales which are well within the credible intervals. This is consistent with the fact that within our approach nucleotide frequencies and substitution rates are assumed to be the same for all trees.

| Edge lengths | Naive-model | True |
|---|---|---|
| $t_1$ | 0.14 (0.02-0.26) | 0.09 |
| $t_2$ | 0.10 (0.03-0.18) | 0.15 |
| $t_3$ | 0.08 (0.02-0.15) | 0.10 |
| $t_4$ | 0.30 (0.13-0.44) | 0.20 |
| $t_5$ | 0.04 (0.01-0.09) | 0.10 |
| $t_6$ | 0.11 (0.01-0.22) | 0.20 |
| $t_7$ | 0.23 (0.02-0.38) | 0.15 |
| $\tilde{t}_8$ | 0.20 (0.04-0.30) | 0.20 |
| $t_9$ | 0.37 (0.10-0.50) | 0.25 |
| $\tilde{t}_{10}$ | 0.11 (0.01-0.29) | 0.25 |
| $t_{11}$ | 0.08 (0.01-0.20) | 0.10 |
| $t_{12}$ | 0.15 (0.04-0.28) | 0.20 |
| $t_{13}$ | 0.40 (0.17-0.50) | 0.30 |
| $t_{14}$ | 0.21 (0.04-0.38) | 0.40 |

Table 5.2: Posterior means ($2.5\%$ and $97.5\%$ quantiles) for the branch lengths when using algorithm (4.9), indicated for convenience by Naive-model, compared to the true branch lengths.

| | Naive-model | True |
|---|---|---|
| Frequencies | | |
| $\pi_A$ | 0.09 (0.04-0.11) | 0.10 |
| $\pi_C$ | 0.42 (0.38-0.46) | 0.40 |
| $\pi_G$ | 0.10 (0.06-0.13) | 0.10 |
| $\pi_T$ | 0.39 (0.30-0.42) | 0.40 |
| Rates | | |
| $r_{AC}$ | 0.10 (0.07-0.15) | 0.09 |
| $r_{AG}$ | 0.21 (0.14-0.28) | 0.22 |
| $r_{AT}$ | 0.14 (0.07-0.21) | 0.14 |
| $r_{CG}$ | 0.16 (0.10-0.25) | 0.16 |
| $r_{CT}$ | 0.36 (0.30-0.44) | 0.35 |
| $r_{GT}$ | 0.03 (0.00-0.06) | 0.04 |

Table 5.3: Posterior means ($2.5\%$ and $97.5\%$ quantiles) for the nucleotide frequencies and rates of substituion when using algorithm (4.9) compared to their true values.

## 5.4 Simulation results using HMM structure

### 5.4.1 Inferring tree topologies

Algorithm (4.12) was run for 600000 iterations with burn-in as described before. Figure 5-4 shows the posterior probabilities for the four topologies against the site $i$ in the multiple alignment, and the means of the posterior probability of **S** for the four regions. Looking at this figure, the probabilities show a very clear signal, and hence the classification performance is pretty good. This is the result of the HMM prior, which accounts for correlations between neighboring sites. The use of the HMM shows a considerably improved classification performance as compared to the naive approach. Notice that the breakpoint estimates are systematically shifted. Algorithm (4.12) was run several times with different $\nu$ fixed at some smaller values, such as $0.90$ and $0.80$. The asymmetry in the breakpoint locations did not disappear. The reason for this could possibly be ascribed to the directionality effect of the HMM structure, as HMMs allow us to model dependencies between sites accounting for one neighbor only.

### 5.4.2 Parameter estimates

Table 5.4 reports the estimates for the fourteen branch lengths. From the results we can see that procedure (4.12) is able to recover the true synthetic parameter

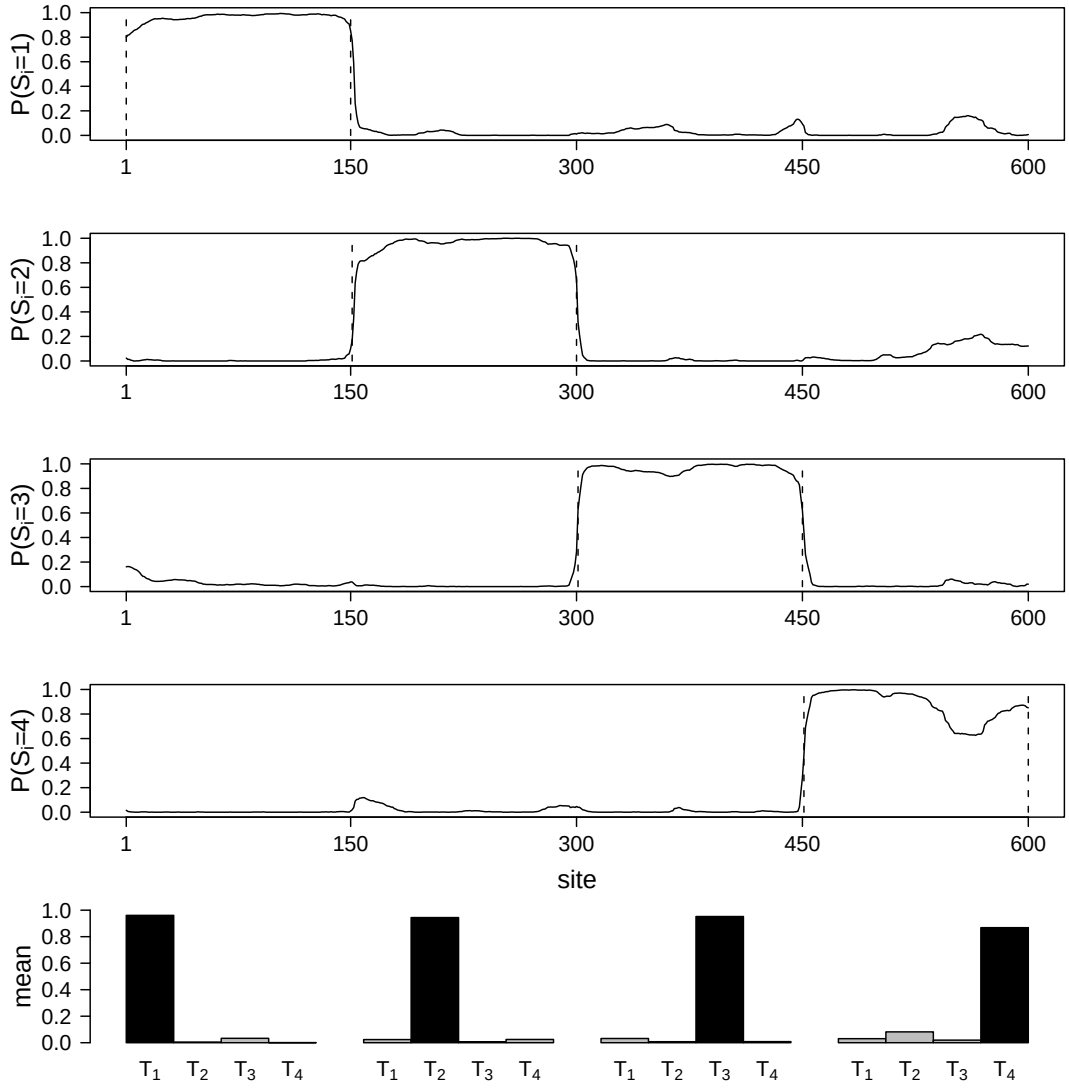



Figure 5-4: Inferred HGTs with HMMs. The first four plots from the top show the posterior probabilities for the four states. The bar plots (bottom) show the mean of the posterior probabilities of **S** for the four topologies in each region. Notice the better performance of the HMM as compared to the naive prior.

values. In fact, the estimates are close to the true vales which are well within the credible intervals. The same considerations apply to the estimates of the nucleotide frequencies, rates of substitution and probability of not changing topology presented in Table 5.5.

Given the significantly better performance, the HMM prior will be used for the rest of the chapter.

| Edge lengths | HMM-model | True |
|---|---|---|
| $t_1$ | 0.07 (0.04-0.10) | 0.09 |
| $t_2$ | 0.14 (0.08-0.20) | 0.15 |
| $t_3$ | 0.09 (0.04-0.14) | 0.10 |
| $t_4$ | 0.21 (0.12-0.30) | 0.20 |
| $t_5$ | 0.09 (0.04-0.15) | 0.10 |
| $t_6$ | 0.19 (0.10-0.28) | 0.20 |
| $t_7$ | 0.17 (0.07-0.26) | 0.15 |
| $\tilde{t}_8$ | 0.20 (0.12-0.29) | 0.20 |
| $t_9$ | 0.27 (0.18-0.40) | 0.25 |
| $\tilde{t}_{10}$ | 0.27 (0.14-0.40) | 0.25 |
| $t_{11}$ | 0.07 (0.03-0.11) | 0.10 |
| $t_{12}$ | 0.23 (0.11-0.31) | 0.20 |
| $t_{13}$ | 0.33 (0.23-0.44) | 0.30 |
| $t_{14}$ | 0.37 (0.24-0.52) | 0.40 |

Table 5.4: Posterior means ($2.5\%$ and $97.5\%$ quantiles) for the branch lengths when using algorithm (4.12), indicated for convenience by HMM-model, compared to the true branch lengths.

Any MCMC algorithm can suffer from two problems: slow or lack of convergence and poor mixing. There are several graphical and statistical methods to check mixing and convergence (Brooks and Gelman, 1998; Cowles and Carlin, 1996). Here we use the time series (or trace) and autocorrelation function plots as well as acceptance rate. A trace plot is a plot of the iteration number against the value of the draw of the parameter at each iteration. A chain that mixes well traverses its posterior space rapidly, and it can jump from one remote region of the posterior to another in relatively few steps. Another way to assess convergence is to assess the autocorrelations between the draws of our Markov chain. We would expect the $k$th lag autocorrelation to be smaller as $k$ increases (our 2nd and 50th draws should be less correlated than our 2nd and 4th draws). If autocorrelation is still relatively high for higher values of $k$, this indicates high degree of correlation between our draws and slow mixing. The acceptance rate is the percentage of accepted proposals. A good rate is between 30% and 70%. These diagnostic tests and many others are available in

| | HMM-model | True |
|---|---|---|
| Frequencies | | |
| $\pi_A$ | 0.10 (0.08-0.12) | 0.10 |
| $\pi_C$ | 0.40 (0.38-0.42) | 0.40 |
| $\pi_G$ | 0.11 (0.10-0.12) | 0.10 |
| $\pi_T$ | 0.39 (0.38-0.40) | 0.40 |
| Rates | | |
| $r_{AC}$ | 0.11 (0.08-0.13) | 0.09 |
| $r_{AG}$ | 0.20 (0.16-0.25) | 0.22 |
| $r_{AT}$ | 0.14 (0.12-0.17) | 0.14 |
| $r_{CG}$ | 0.16 (0.13-0.19) | 0.16 |
| $r_{CT}$ | 0.36 (0.33-0.39) | 0.35 |
| $r_{GT}$ | 0.03 (0.01-0.05) | 0.04 |
| Prob. of no HGT | | |
| $\nu$ | 0.99 (0.97-1.00) | 0.99 |

Table 5.5: Posterior means ($2.5\%$ and $97.5\%$ quantiles) for the nucleotide frequencies, rates of substituion and probability of not changing topology when using algorithm (4.12) compared to their true values.

R (e.g. CODA package). The plots, in Figures 8-1–8-4 in the Appendix, indicate that adequate mixing and convergence is achieved for all parameters; for all the trace plots the center of the chain appears to be around the true value, with very small fluctuations, and the chain is exploring the distribution by traversing to areas where its density is very low. Also, the autocrrelation plots drop to very small values as the time lag increases. The acceptance rates are between 40% and 65%. Not surprisingly, the autocorrelation function and trace plots for the naive case (not reported here) are worse as the uniform prior fails to capture the dependencies between adjacent sites.

### 5.4.3 Gibbs sampling versus forward-backward algorithm

Before illustrating the difference in convergence and mixing between the Gibbs sampling scheme and forward-backward algorithm MCMC sampler, it is useful to introduce the concept of integrated autocorrelation time which provides a means for assessing the performance of a method in comparison to independent sampling (Green and Han, 1992). This is often indicated by $\tau(f)$. If $\tau(f) < 1$ ($\tau(f) > 1$) then the chosen method performs better (worse) than independent sampling ($\tau(f) = 1$). Here we use $\tau(f)$ to compare different MCMC algorithms; the smaller the $\tau(f)$ the better the method in terms of accuracy of estimation. Note that when comparing methods, one should also consider

computational cost. $\tau(f)$ can be expressed as

$$\tau(f) = \sum_{t=-\infty}^{\infty} \rho_t(f) \tag{5.1}$$

where $\rho_t(f)$ is the lag $t$ autocorrelation of a stationary chain. We follow the approach by Geyer (1992) to estimate (5.1). That is,

$$\hat{\tau}(f) = -1 + 2\sum_{i=0}^{G} \hat{\Gamma}_i \tag{5.2}$$

where $\hat{\Gamma}_i = \hat{\rho}_{2i} + \hat{\rho}_{2i+1}$ is the sum of adjacent pairs of sample autocorrelations and $\hat{\rho}_t$ is the autocorrelation at lag $t$. Here $G$ is chosen to be the largest integer such that $\hat{\Gamma}_i > 0$, for $i = 0, 1, \ldots, G$. Estimator (5.2) is known as the initial positive sequence estimator.

The integrated autocorrelation time gives information about the correlation structure of a chain; the smaller the $\tau(f)$, the smaller the correlation between the samples. According to Geyer (1992), $\tau(f)$ can in fact be used as a means to assess the mixing features of a chain. In particular, a chain which moves quickly around the support of the target distribution (i.e. mixing is quick) will have a smaller $\tau(f)$ as compared to when a chain moves slowly. In this respect, more reliable estimates are obtained when a chain mixes rapidly as compared to the case in which it mixes slowly. Obviously the former situation is preferable as long as the computational cost associated to the rapidly-mixing chain is not prohibitive.

To illustrate the difference in convergence and mixing between the Gibbs sampling scheme and the forward-backward algorithm MCMC sampler was run for 15000 iterations with burn-in equal to 20. The resulting posterior probabilities and means of these probabilities for the four topologies are depicted in Figures 5-5 and 5-6. For illustration purposes, we also provide the autocorrelation function and the trace plots for $S_{50}$, $S_{350}$ and $S_{500}$ of both procedures shown in Figures 5-7 and 5-8. Table 5.6 reports the estimated integrated autocorrelation times for the averaged sequence of tree topologies. Figures 5-5 and 5-6 show that the forward-backward algorithm outperforms the Gibbs sampling in terms of posterior probabilities and hence tree allocation. Figure 5-7 highlights the very high parameter autocorrelations when using the Gibbs sampler as compared to the forward-backward algorithm (see Figure 5-8). The estimated integrated autocorrelation time of the forward-backward algorithm is only 42 (see Table 5.6), which makes this method the one with the best estima-

tion performance, relative to the Gibbs sampling. Table 5.6 also shows that the forward-backward algorithm took nearly as long to run as the Gibbs sampling. Hence, we can conclude that the performance of the Gibbs sampler is poor as compared to that of the forward-backward algorithm. In fact, while the Gibbs sampler requires a considerable higher burn-in and suffers from very high parameter autocorrelations, the forward-backward approach requires very little burn-in and convergence is achieved in about the same amount of time as the Gibbs sampler. The forward-backward algorithm will be used throughout the rest of the chapter.

| | $\hat{\tau}$ | time in hours |
|---|---|---|
| GS | 279 | 2.43 |
| SFBA | 42 | 2.58 |

Table 5.6: The estimated integrated autocorrelation time ($\hat{\tau}$) and the computational cost (measured as execution time in hours) for the Gibbs sampler (GS) and the stochastic forward-backward algorithm (SFBA).

### 5.4.4 Different tree topology structures

We know that a phylogenetic network can be decomposed into trees. It may happen that trees within a network can be more similar and hence more difficult to differentiate from data. This can cause distortion and instability of the results, particularly of the posterior probabilities of the sequence of topologies, and ultimately affect the tree classification performance. To illustrate this point consider DNA sequences, 600 bases long, which are evolved under two different scenarios, using the GTR model with same nucleotide frequencies and rates of substitution as described in Section 5.2.

- Scenario 1. The DNA sequences are evolved along the network drawn in Figure 5-9, with branch lengths equal to $(0.1, 0.2, 0.2, 0.4, 0.1, 0.1, 0.1, 0.1)$ and hence along the tree represented in Figure 5-10a between nucleotides $i = 1-300$ and the tree in Figure 5-10b between nucleotides $i = 301-600$.

- Scenario 2. The first 300 DNA sequences are evolved along the tree in Figure 5-12a and the last under the tree in Figure 5-12b, both induced by the network in Figure 5-11 with edges $(0.2, 0.2, 0.1, 0.1, 0.1, 0.2, 0.3, 0.1)$.

The only difference between the two scenarios is where the horizontal gene transfer is occurring. In fact, in scenario 1, Taxon 1 is horizontally transferring genetic material to Taxon 3, whereas in scenario 2, Taxon 4 is laterally

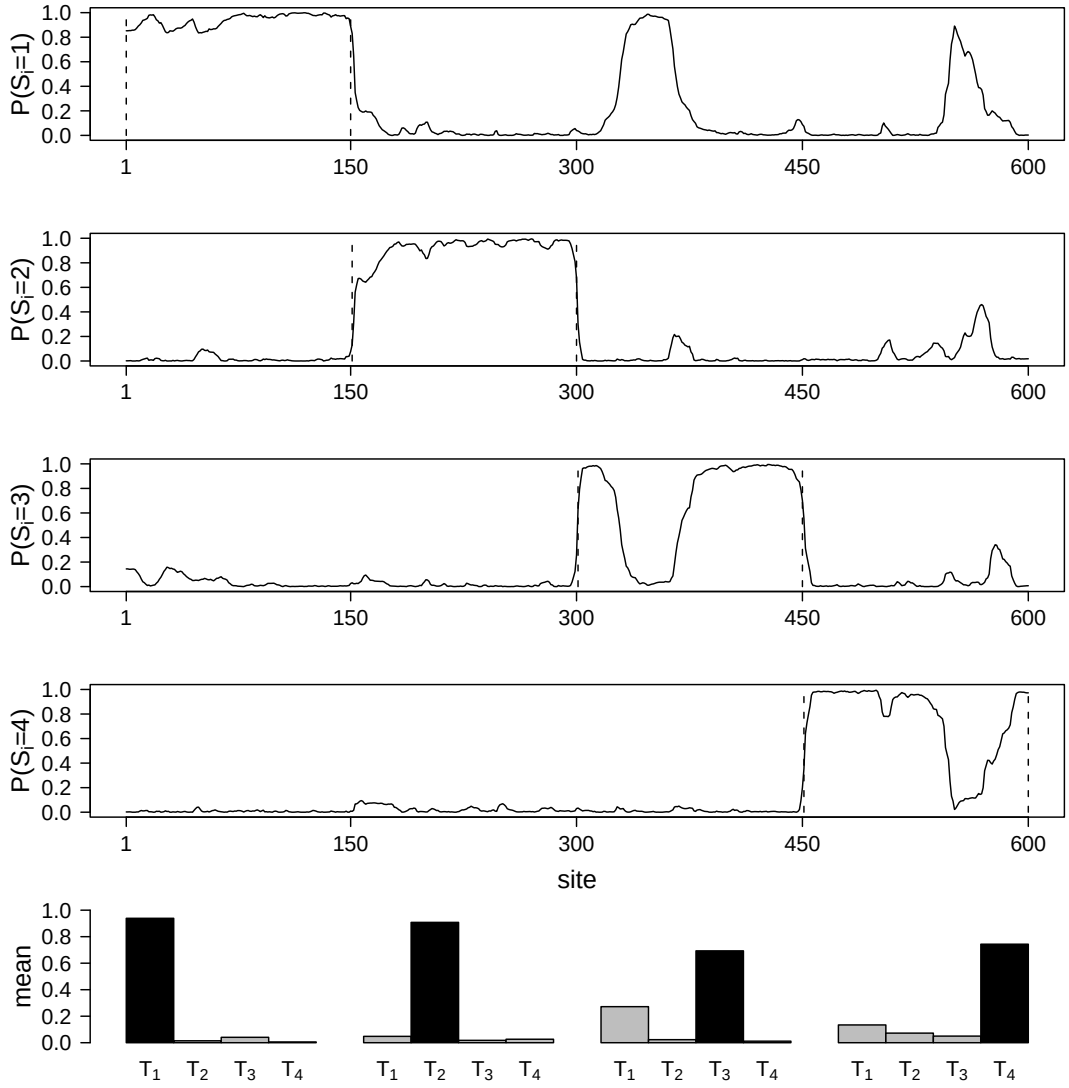



Figure 5-5: Inferred HMM approach with Gibbs sampler. The first four plots from the top show the posterior probabilities for the four tree topologies. The bar plots (bottom row) show the mean of the posterior probabilities of **S** for the four topologies in each region.

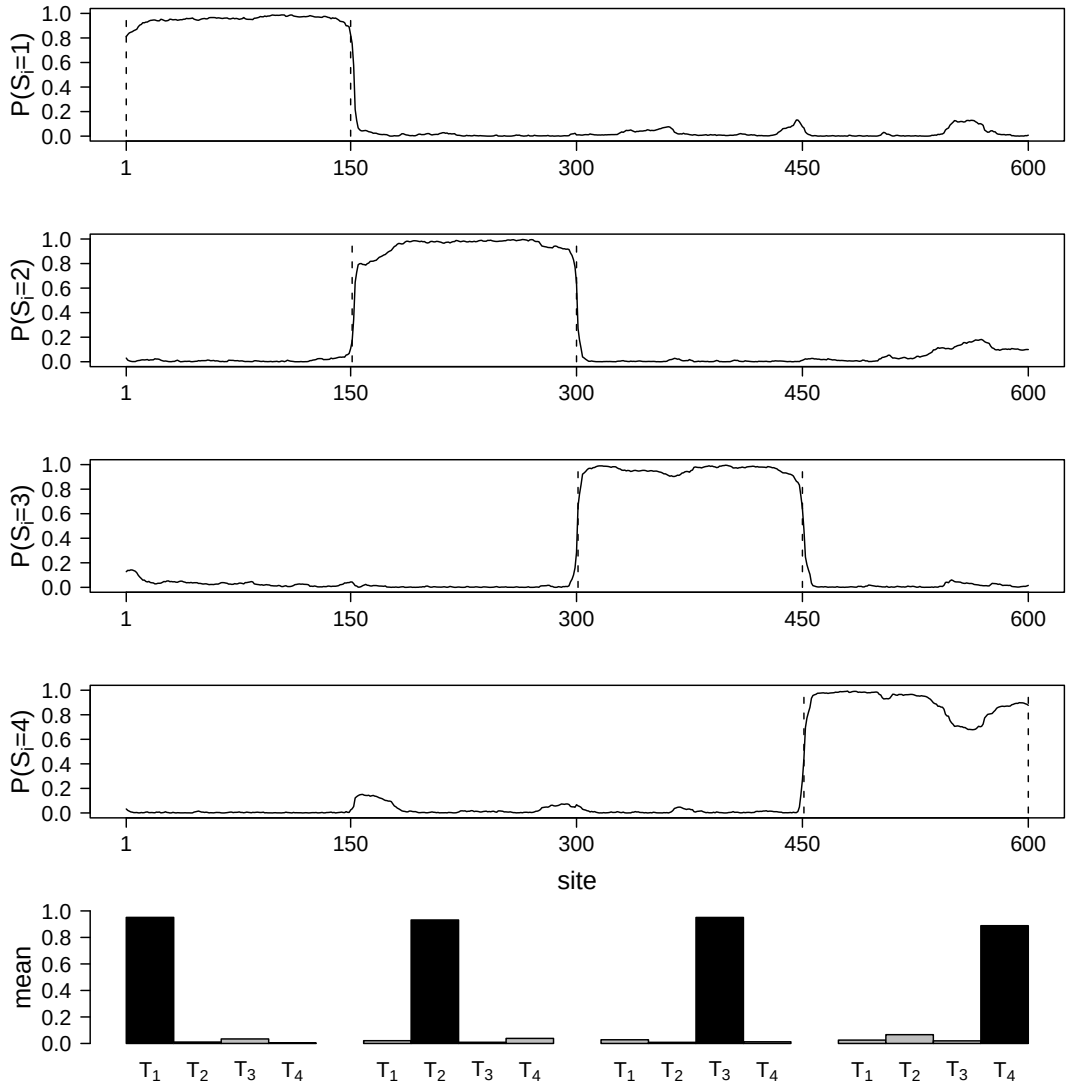



Figure 5-6: Inferred HMM approach with forward-backward algorithm. The first four plots from the top show the posterior probabilities for the four states. The bar plots (bottom) show the mean of the posterior probabilities of **S** for the four topologies in each region. Notice the better performance of this algorithm as compared to the Gibbs sampler.

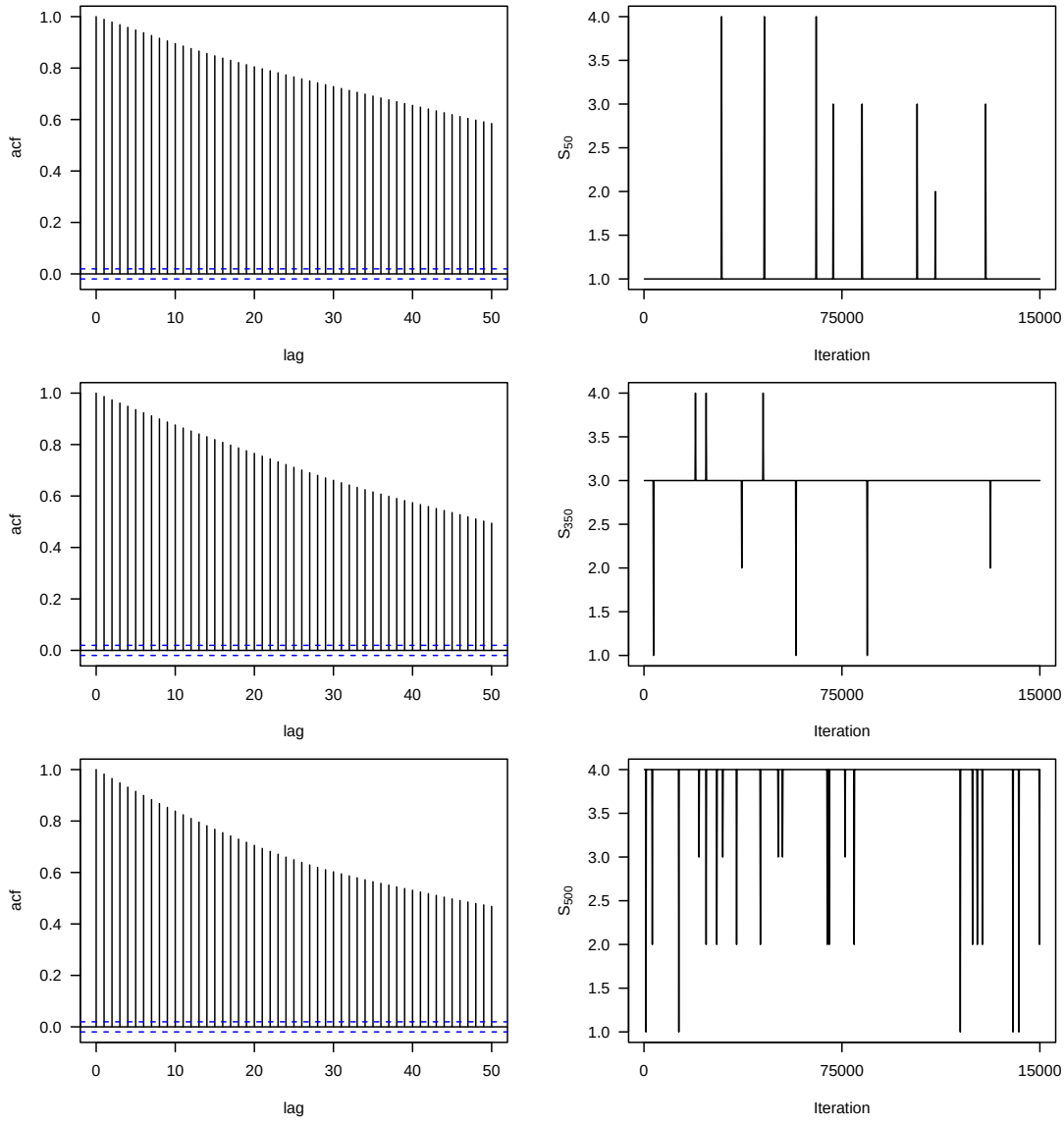



Figure 5-7: Autocorrelation function and trace plots for $S_i,\ i = 50, 350, 500$ with Gibbs sampling algorithm.

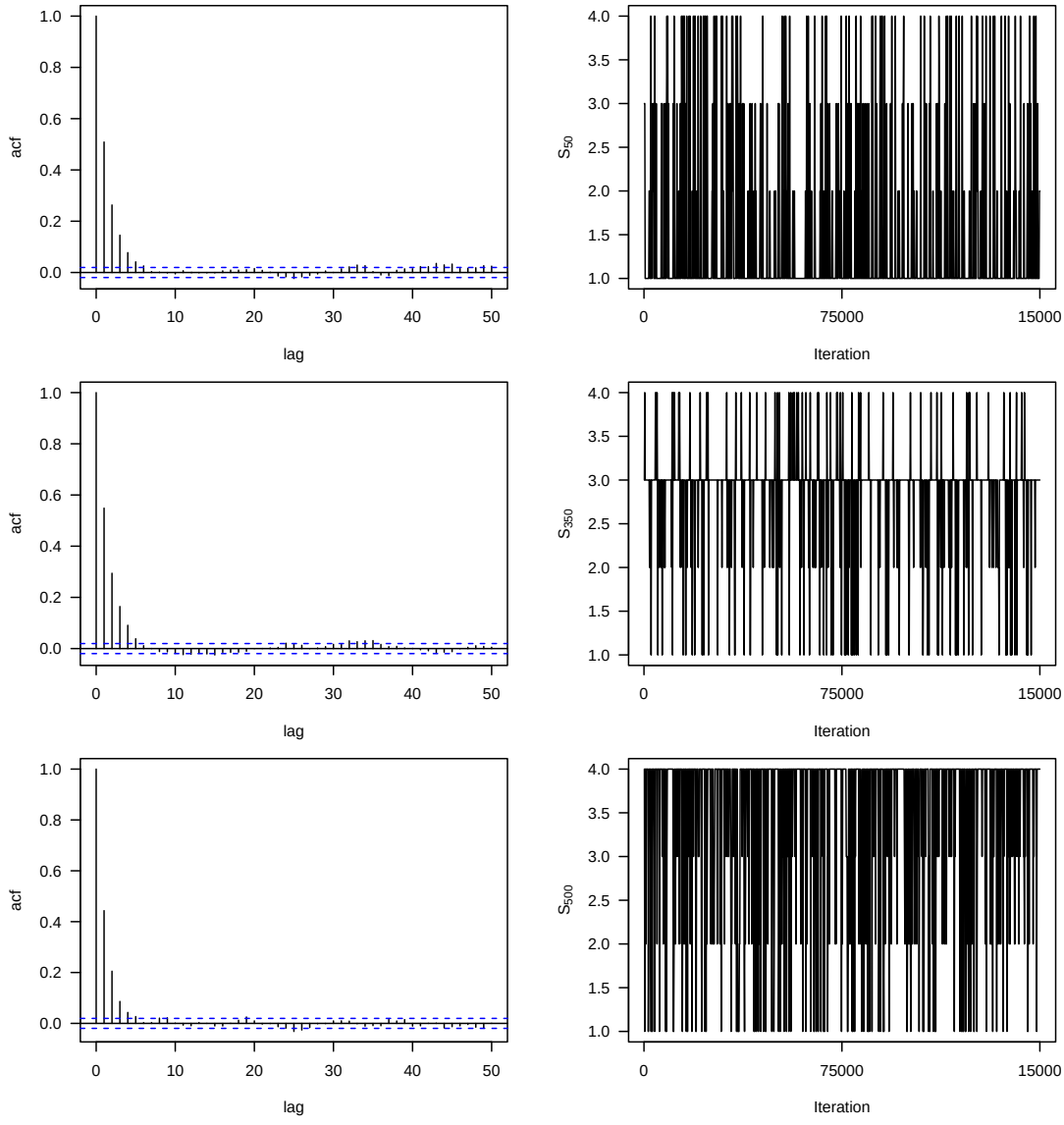



Figure 5-8: Autocorrelation function and trace plots for $S_i,\ i = 50, 350, 500$ with stochastic forward-backward algorithm. Notice the better performance of this algorithm in terms of mixing and convergence as compared to the Gibbs sampler.

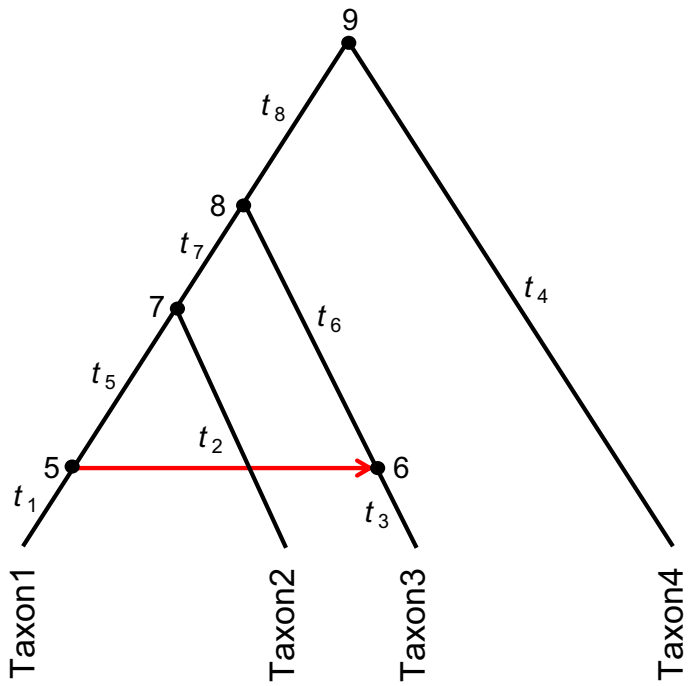
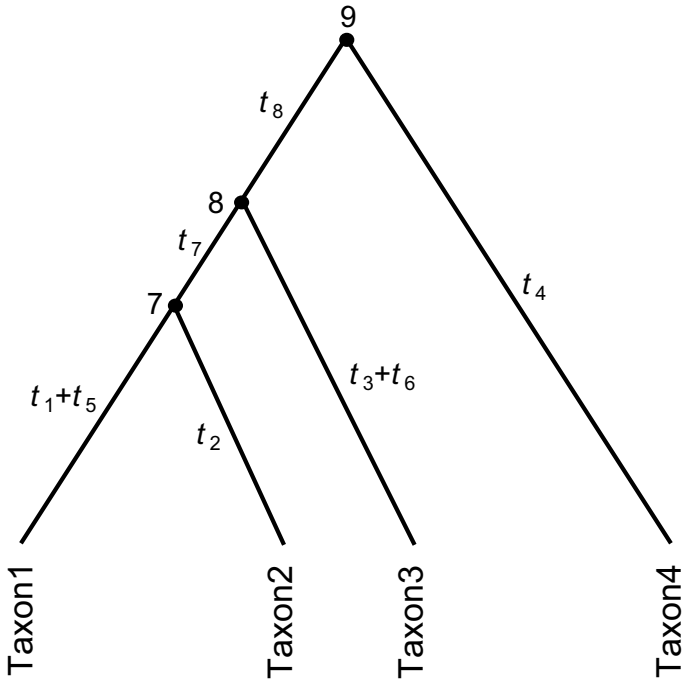


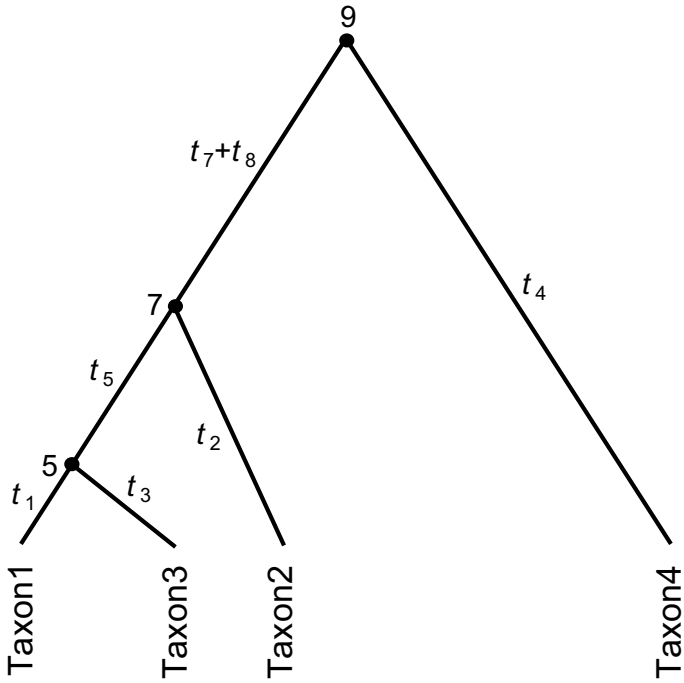

Figure 5-9: Scenario 1. Phylogenetic network of four taxa with one reticulation event (R=1) and eight branch lengths.

transferring DNA material to Taxon 3. This means that the underlying species trees for the two networks are the same as shown in Figures 5-10a and 5-12a whereas the horizontally transferred trees are not (Figures 5-10b and 5-12b). The MCMC algorithm was run for 15000 iterations with the first 100 discarded as burn-in. By looking at the posterior probabilities in the first two top rows of Figures 5-13 and 5-14 it can be easily seen that the values under scenario 1 are more noisy than those under scenario 2. Also the barplots under scenario 1 (bottom row of Figure 5-13) show a poorer classification as compared to that of scenario 2 (bottom row of Figure 5-14).

To understand the reason for this, let us look in detail at the four trees. Comparing the two trees within scenario 1, we see that their structure is more alike as compared to that of trees in scenario 2. This means that in the presence of little DNA site variability, under scenario 1, $T_1$ is less distinguishable from $T_2$ (except for the branch lengths). Therefore algorithm (4.12) discriminates less well between the two trees as compared to the case of scenario 2 where the trees are more distinguishable, hence leading to better results in terms of probabilities and tree classification.




(a) The underlying species $T_1$ that does not include the reticulation edge under scenario 1.




(b) The horizontally transferred gene tree $T_2$ that includes the reticulation edge under scenario 1.

Figure 5-10: Trees induced by the network in Figure 5-9.

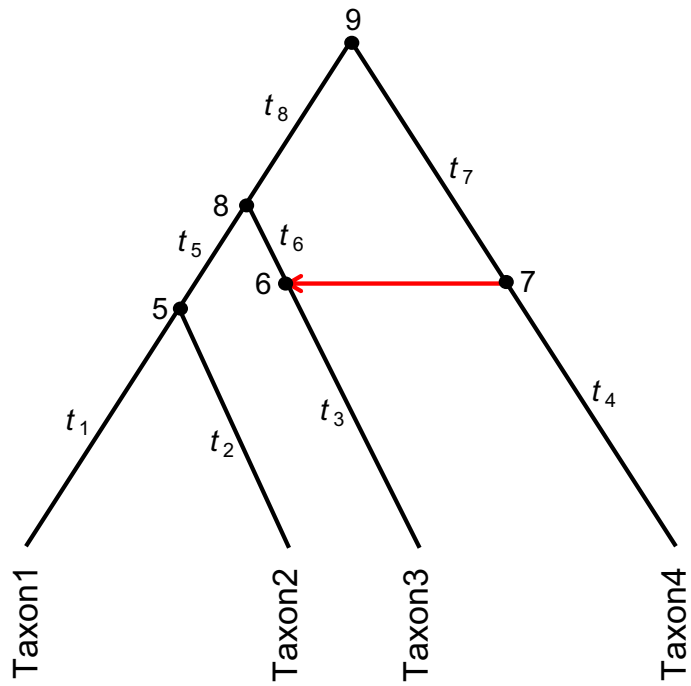
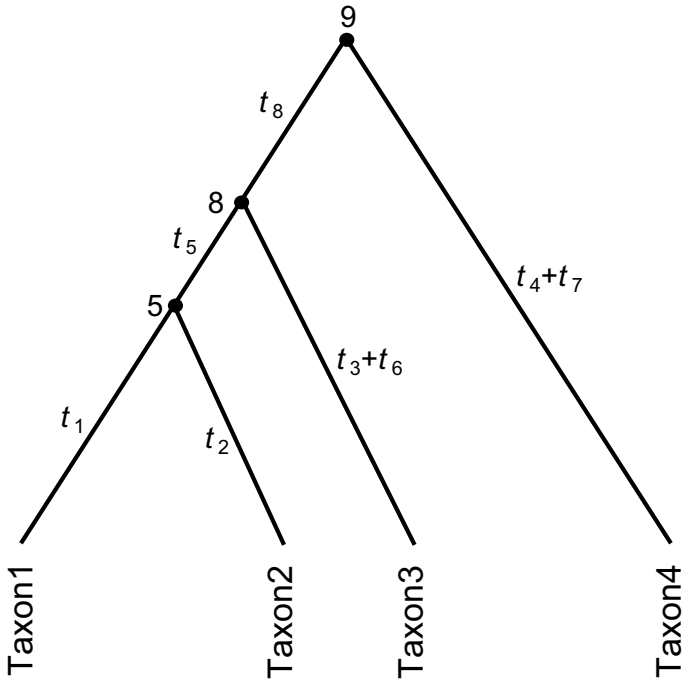



Figure 5-11: Scenario 2. Phylogenetic network of four taxa with one reticulation event (R=1) and eight branch lengths.

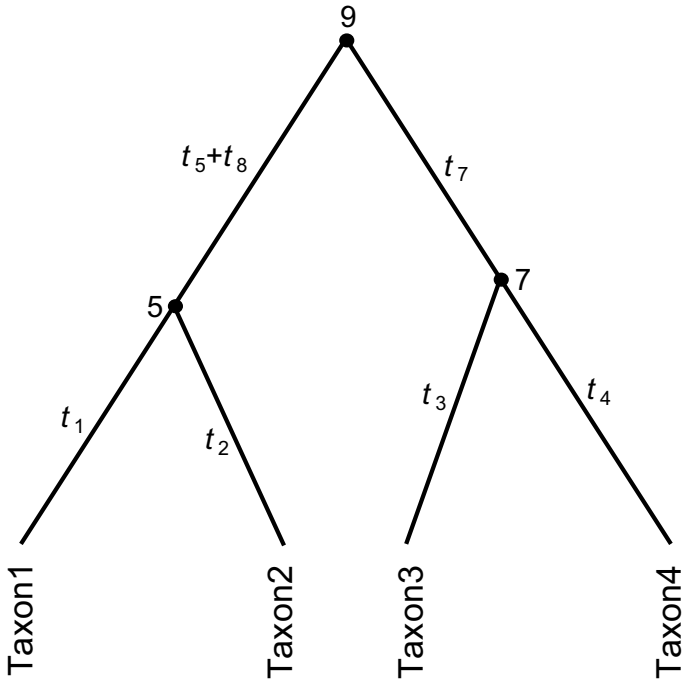
### 5.4.5 Some model misspecifications

The proposed method, as any other, is based on some assumptions. For example we assume that the stochastic model from which the data have been generated, the number of reticulation events and hence the phylogenetic network topology, are known. As in Jin *et al*. (2006) these are working assumptions which might not be satisfied in the real world. So here, we assess the impact of violation of these assumptions on the posterior probabilities and classification of the tree topologies. To this end, the following three scenarios are considered. For all cases algorithm (4.12) was run for 15000 iterations with the first 100 discarded as burn-in.

- Misspecification 1. Data are generated as in Section 5.2 and the parameters of interest estimated by using the Jukes Cantor model (2.4), which is rather simple as compared to the GTR. The plots reported in Figure 5-15 show that this misspecification affects the classification of the tree topologies in a mild way. Indeed there are some spurious spikes in the posterior probabilities but overall the pattern is clear and tree classification acceptable: the choice of the model of evolution is important but not crucial, unless we are directly interested in the nucleotide frequencies and rates of substitution. This means that simpler models can still be employed




(a) The underlying species $T_1$ that does not include the reticulation edge under scenario 2.




(b) The horizontally transferred gene tree $T_2$ that includes the reticulation edge under scenario 2.

Figure 5-12: Trees induced by the network in Figure 5-11.

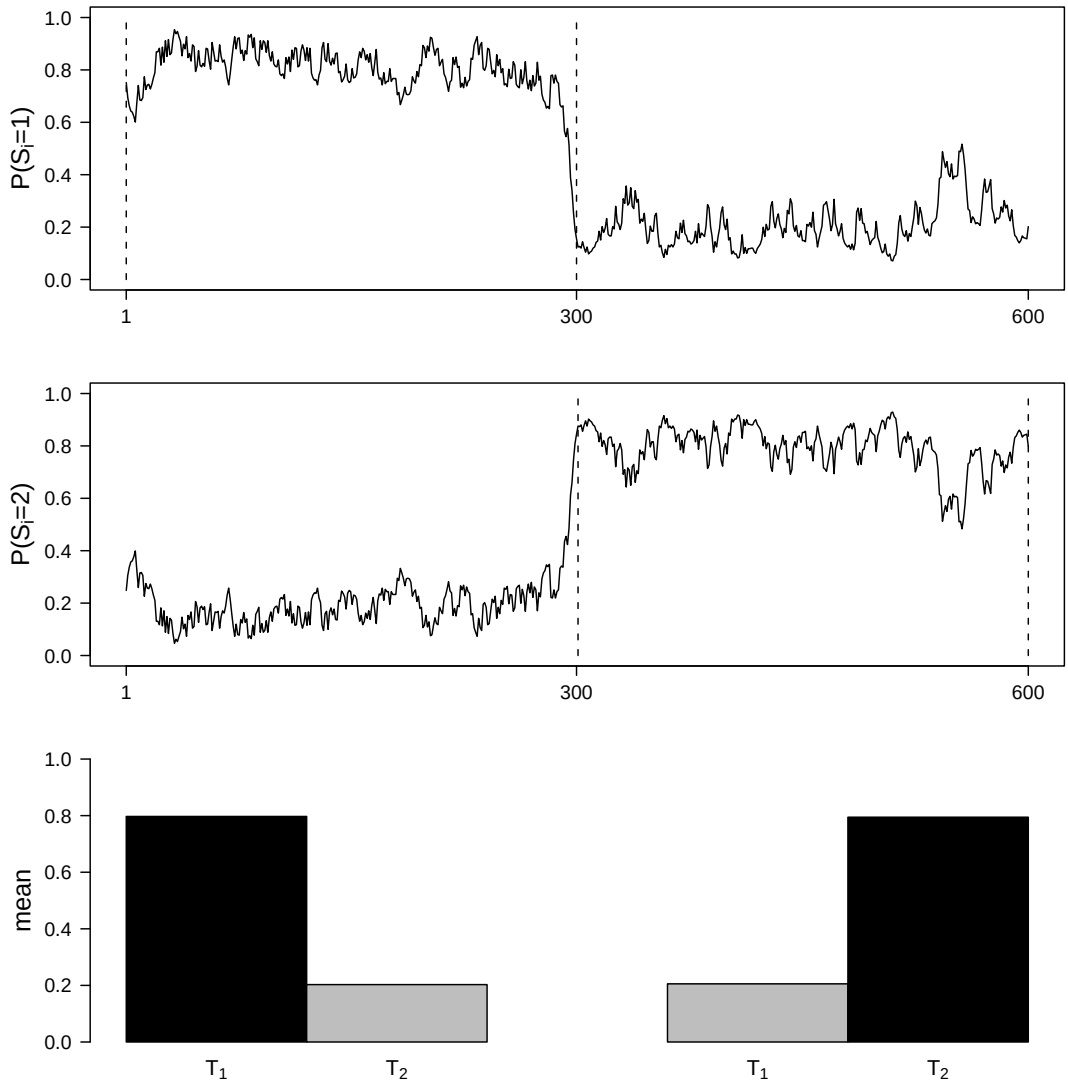



Figure 5-13: Posterior probabilities for the two tree topologies (first two top rows) and bar plots (bottom) showing the mean of the posterior probabilities of **S** for $T_1$ and $T_2$ in each region under scenario 1.

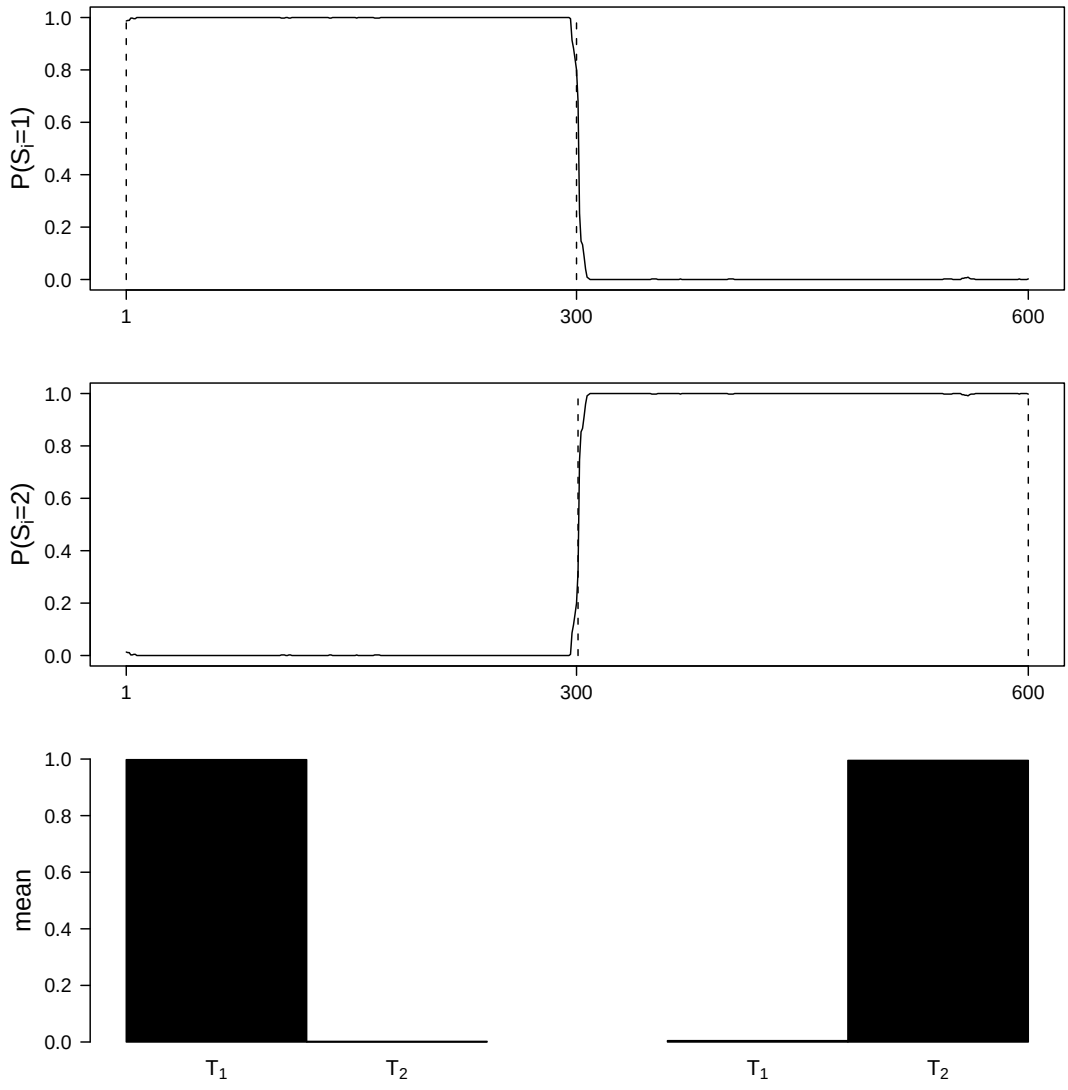



Figure 5-14: Posterior probabilities for the two tree topologies (first two top rows) and bar plots (bottom) showing the mean of the posterior probabilities of **S** for $T_1$ and $T_2$ in each region under scenario 2.

for classification purposes, therefore decreasing the running time of the method.

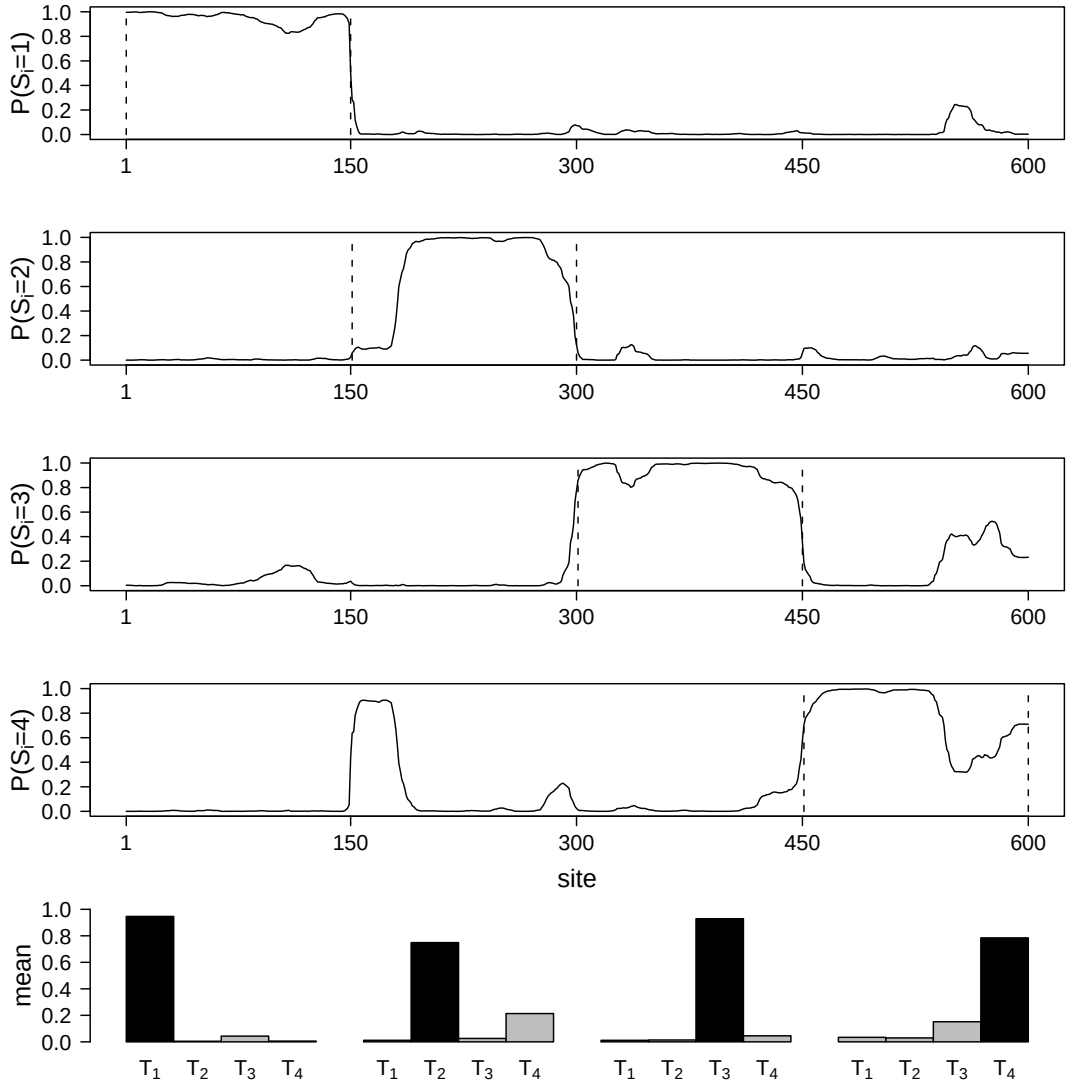



Figure 5-15: Misspecification 1. The four top row plots show the posterior probabilities for the four tree topologies. The bar plots (bottom) show the mean of the posterior probabilities of **S** for the four topologies. The data are generated under the GTR model but the parameters estimated using a Jukes Cantor model. Notice the weak effect of this misspecification on the tree allocations.

- Misspecification 2. The DNA sequences are evolved along the network represented in Figure 5-1 but with just one HGT represented by edge (9,10), that is the data are generated under $T_1$ in Figure 5-2a for the sites (1-300) and under $T_2$ in Figure 5-2b for the remaining sites (301-600), using the GTR model of Section 5.2. The parameters are estimated by using

a GTR model but under the network described in Figure 5-1, hence under trees $T_1$, $T_2$, $T_3$ and $T_4$. The results, reported in Figure 5-16, show that having more trees than those required by the data generating process (DGP) is not problematic because the algorithm will give small or zero probability to the nuisance trees, hence just select the most appropriate ones. This means that if we are unsure about the number of reticulation events, we can specify a more complex model with more HGTs and then let the algorithm choose the trees with highest probabilities. This problem can be thought of as a kind of variable (tree) selection where only the most important variables are chosen. However, given a species tree, for high dimensional tasks, the use of all possible tree combinations will impose a greater computational cost. In this case, more efficient numerical algorithms are needed. This point is addressed in Chapter 7.

- Misspecification 3. The data are generated with the GTR model described in Section 5.2 using the network in Figure 5-9. The parameters are estimated using a different network (see Figure 5-17) and hence using the trees reported in Figure 5-18a and 5-18b. The posterior probability distribution of the $S_i$ and tree classification are reported in Figure 5-19. When the network of the DGP is different from the network used in estimation, the resulting probabilities are noisy but most importantly no tree can be preferred over another. In other words, if the location of the HGTs is wrongly placed on the species tree, then this is substantially reflected in the bad posterior probabilities and tree classification. If such a case occurs in applied work, then we can be confident that the chosen network is wrong.

## 5.5 Discussion

The MCMC procedure has been applied to synthetic data showing that this algorithm is able to recover the true synthetic parameter values.

The naive classifier has been contrasted to the HMM for the sequence of topologies. The results have shown that as different sites are modelled independently, a weak signal at a certain site will cause the inference of an erroneous tree at that site. The application of the HMM redeems this deficiency.

As for the Gibbs sampling and forward-backward algorithm comparison, not surprisingly, the performance of the former is poor. In contrast the performance of the latter is much better, requiring very little burn-in as convergence

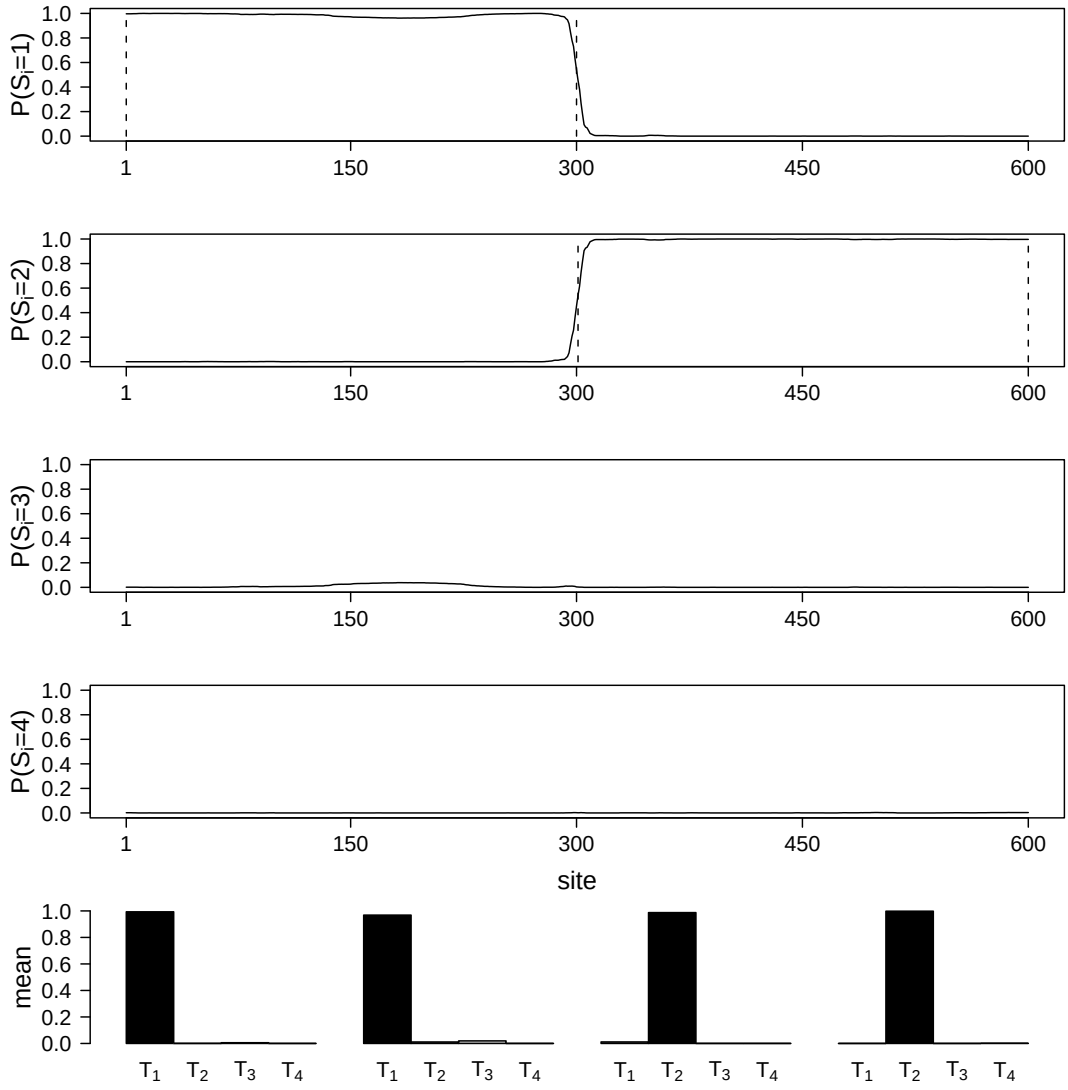



Figure 5-16: Misspecification 2. The four top row plots show the posterior probabilities for the four tree topologies. The bar plots (bottom) show the mean of the posterior probabilities of **S** for the four topologies. The data are generated via GTR model using just two trees (one reticulation event) but parameters estimated with four tree topologies (two reticulation events).

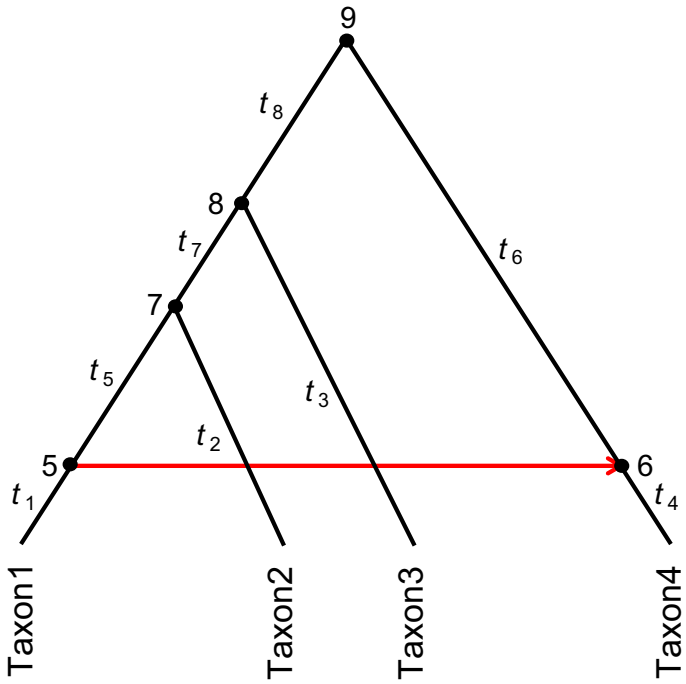
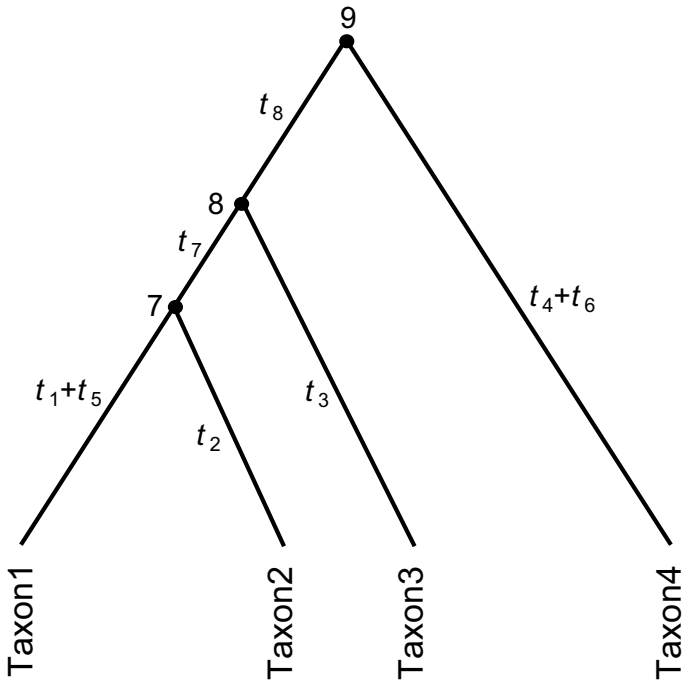
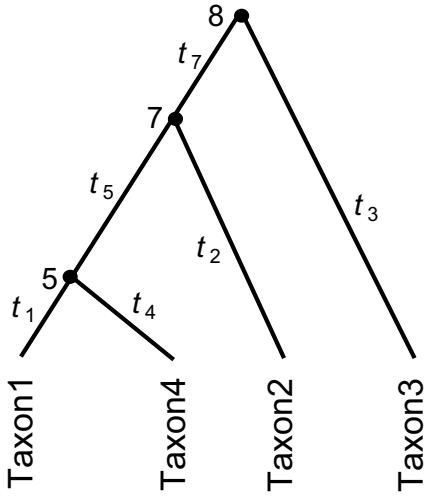



Figure 5-17: The Phylogenetic network of four taxa with one reticulation event (R=1) and eight branch lengths used in the estimation procedure under misspecification 3.




(a) The underlying species that does not include the reticulation edge used in the estimation procedure under misspecification 3.




(b) The horizontally transferred gene tree that includes the reticulation edge $(5, 6)$ used in the estimation procedure under misspecification 3.

Figure 5-18: Trees induced by the network in Figure 5-17.

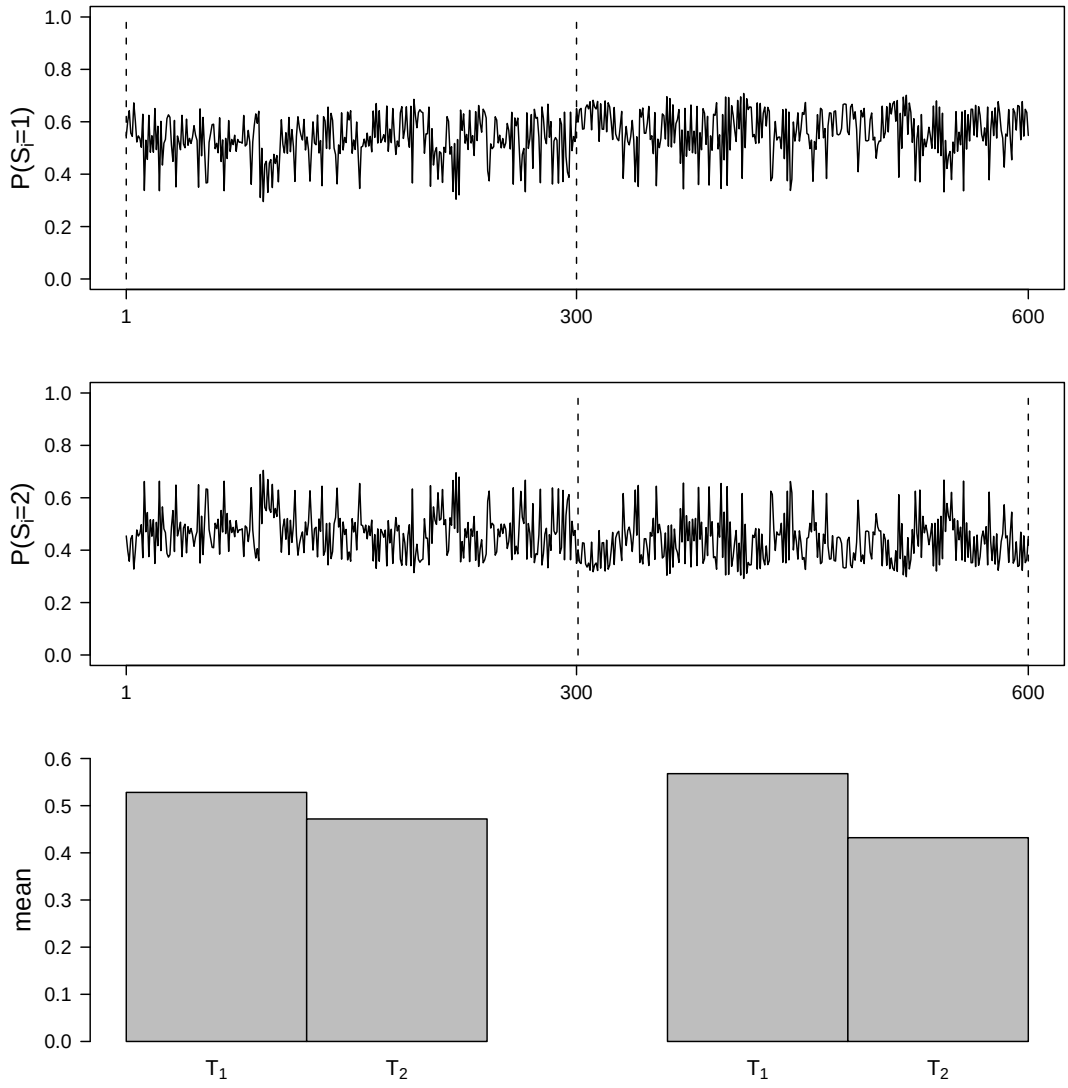



Figure 5-19: Misspecification 3. The two top row plots show the posterior probabilities for the two tree topologies. The bar plots (bottom) show the mean of the posterior probabilities of **S** for the two topologies. The data are generated with a GTR model using the network in Figure 5-9 but the parameters estimated with the network described in Figure 5-17.

is achieved quickly.

Also several scenarios and misspecification cases have been presented with the aim of obtaining some operational insights: (1) specifying the location where HGTs occur is important as it produces different tree topologies that the MCMC algorithm might or might not discriminate well; (2) depending on the problem at hand the choice of the model of evolution may be not crucial as acceptable tree classifications can be obtained by using simpler models; (3) the MCMC algorithm could be generalised to do tree selection relaxing the assumptions that the number and the position of the HGTs are known *a priori*. This is known to be a very challenging task; (4) if HGTs are wrongly placed on the species tree this might be reflected in bad posterior probabilities and tree classification.

# Chapter 6

# Analysis of the ribosomal protein gene $rps11$ of flowering plants

## 6.1 Introduction

Here we apply the method of the previous chapter to real data. Specifically, we first introduce the concept of HGT in plants, explain the reason why this is an interesting and important topic, particularly for biotechnologists, and describe the dataset. Then we estimate all the quantities of the phylogenetic network and show some connections with the simulation results of Chapter 5, explaining why some results are more plausible than others. Finally, we briefly discuss the development of a more flexible algorithm that will be the focus of the following Chapter 7.

## 6.2 Horizontal gene transfer in plants

Genome sequencing has revealed that HGT is fairly common and important in certain unicellular microorganisms, bacteria in particular. However the prevalence and importance of HGT in the evolution of multicellular organisms remains unclear. Recent studies indicate that plant DNAs are unusually active in HGT. Specifically, the results of the studies by Bergthorsson *et al*. (2003, 2004) established for the first time that conventional genes are subject to evolutionary HGT during plant evolution, and provided the first unambiguous evidence that plants can donate DNA horizontally to other plants. Artefacts of DNA contamination or mislabelled samples, always a concern when invoking HGT, could be ruled out in the HGT cases they found as multiple sampling showed the results to be entirely reproducible. For several reasons,

they believe the cases reported are merely the tip of a large iceberg of HGT in plants. They suggest viruses, bacteria, fungi, insects, pollen, even meteorites and grafting as vectors for this exchange.

These biological insights are emerging at a time when artificial transfer of genes into food crops remains controversial and often resisted with the claim that it would never occur in nature. Accurate knowledge of this aspect of plant evolution is important to the ongoing controversy about transgenic plants produced by biotechnology.

We use the method described in Chapter 4 for analysing a biological dataset whose evolution includes HGTs. The dataset consists of the ribosomal protein gene $rps11$ of a group of flowering plants which was first analysed by Bergthorsson *et al*. (2003) who suggested that this genetic material underwent HGTs. We investigate a subset of the ribosomal protein gene $rps11$ data which consists of five DNA sequences each 350 bases long. The data are available from GenBank (`http://www.ncbi.nlm.nih.gov/genbank/`) with the following identifiers (accession numbers in square brackets): Cabomba [AY293024], Tradescantia [AY293043], Annona [AY293025], Platanus [AY293033], Abelia [AY293009].

## 6.3 Network set up

When using the method described in Chapter 4, the assumptions that the species tree, position and number of reticulation events are known are required. The species tree for this application is reconstructed from various sources of biological evidence (e.g. Bergthorsson *et al*., 2003; Snir and Tuller, 2009) and the edges between the tree branches rebuilt based on the work of Bergthorsson *et al*. (2003). This enables us to construct the phylogenetic network for the five taxa with $R = 2$ HGT events depicted in Figure 6-1, which induce the trees presented in Figure 6-2. It is important to notice the number of trees implied by the network. This network could induce up to $4$ trees but as already alluded to in Section 2.3, in some cases the number of possible trees in the network is less than $2^R$. This is one of such case. In fact the trees within the network are two instead of four. The justification for this is that: (1) it is unrealistic that the two HGT events will occur together since researchers believe that it is unlikely that a taxon (Annona in this case) is transferring and receiving genetic material at the same time; (2) as shown by Snir and Tuller (2009), the resulting most likely paths do not include the species tree because the amount

of DNA available only contains the regions with HGTs. In general, however, is not easy to select a subsample of trees from the $2^R$ trees. Like here, previous findings and/or biological reasoning could give some directions for the selection. Also, some fast and low-resolution methods, such those described for the split networks, could be used to identify a subset of trees.

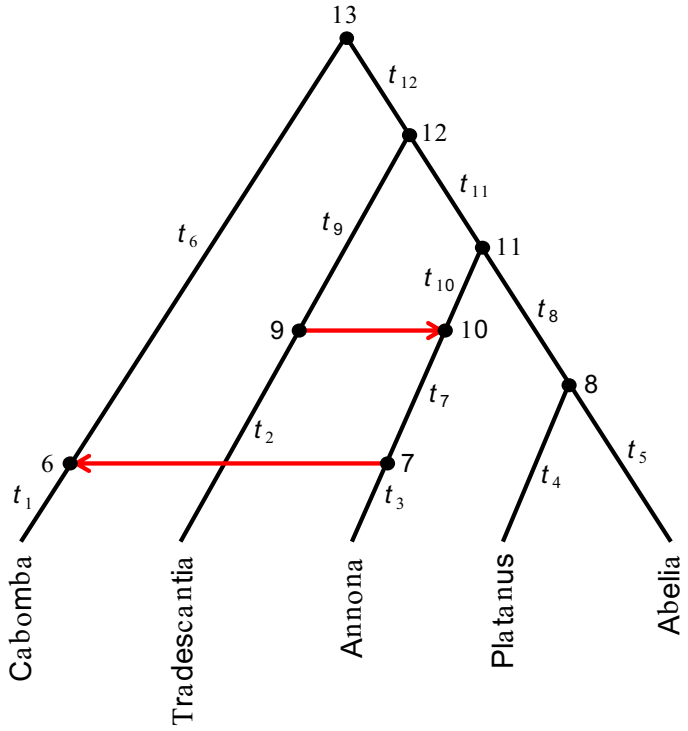
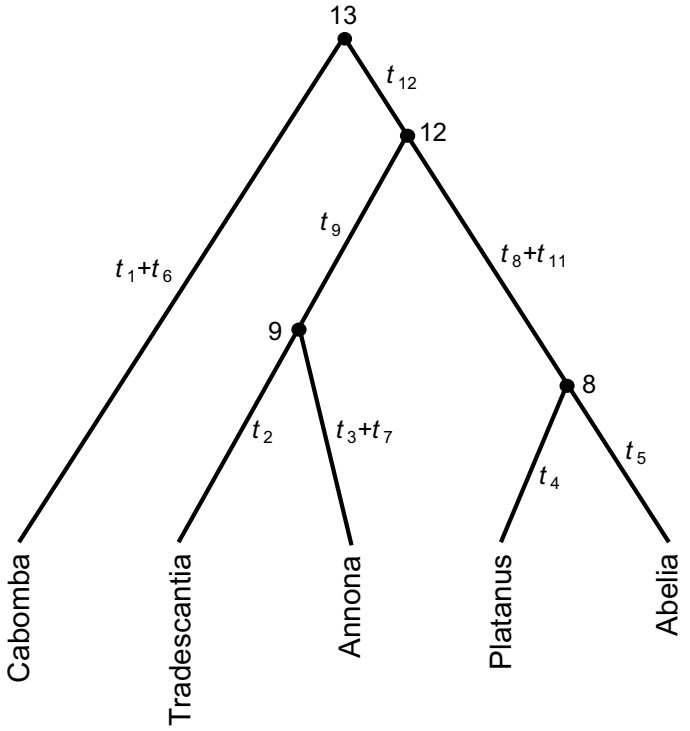



Figure 6-1: Phylogenetic network of the ribosomal protein gene $rps11$ data with $R = 2$ reticulation events: (7,6) and (9,10).

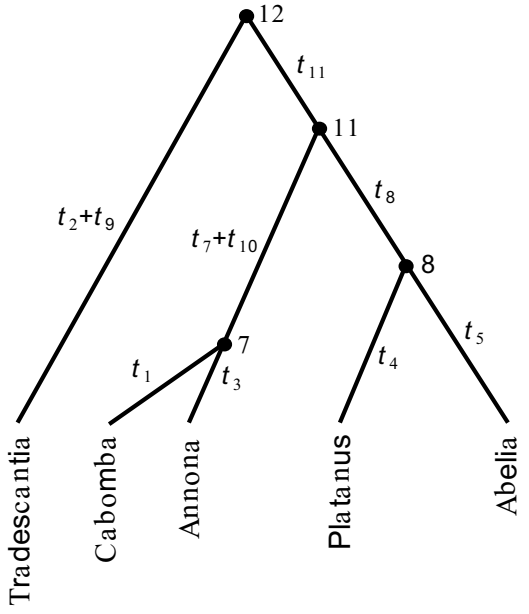
## 6.4 Results using naive approach

### 6.4.1 Inferring tree topologies

Algorithm (4.9) with GTR model was run for 600000 iterations with the first 100000 discarded as burn-in. It is worth recalling that in this algorithm the DNA sites are modelled independently. The results of this approach for the sequence of tree topologies are reported in Figure 6-3 where the probabilities appear to be quite noisy and the mean of the posterior probabilities of **S** for $T_1$ and $T_2$ does not allow us to obtain a satisfactory tree allocation. This is in line with the simulation results shown in Figure 5-3 of Chapter 5, where the performance of the naive classifier is rather poor.




(a) The horizontally transferred gene tree $T_1$ including the edge (9,10).




(b) The horizontally transferred gene tree $T_2$ including the edge (7,6).

Figure 6-2: Trees induced by the network in Figure 6-1.

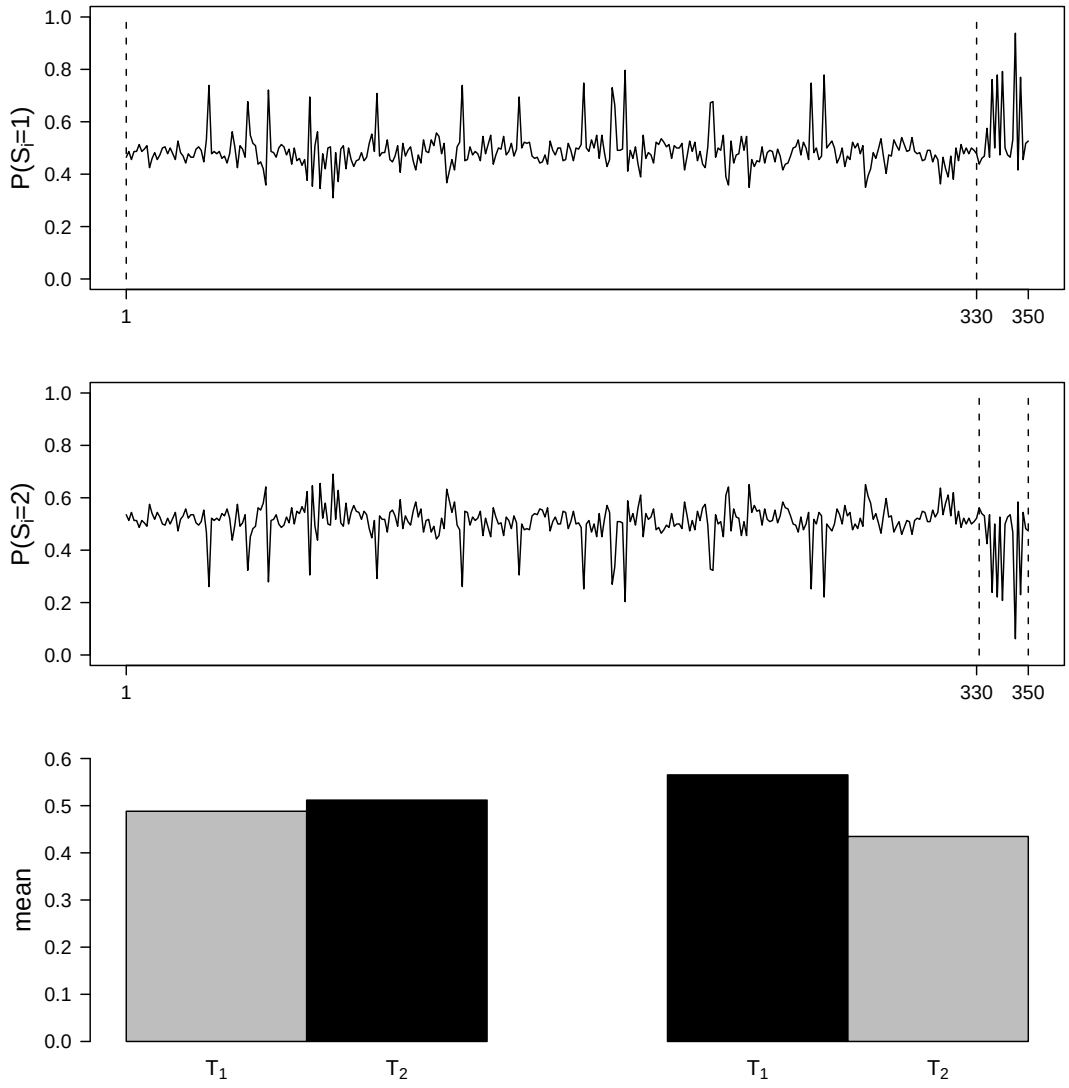



Figure 6-3: Inferred HGTs with naive approach for the ribosomal protein gene $rps11$ data. The first two plots from the top show the posterior probabilities for the two tree topologies (indicated for simplicity by $P(S_i = k),\ k = 1, 2$). The bar plots (bottom row) show the mean of the posterior probabilities of **S** for the two topologies in each region.

### 6.4.2 Parameter estimates

Tables 6.1 and 6.2 contain the estimates of the twelve branch lengths and evolution model parameters. Notice that $\tilde{t}_6 = t_1 + t_6$, $\tilde{t}_7 = t_3 + t_7$, and $\tilde{t}_{10} = t_{10} + \tilde{t}_7 - t_3$.

| Edge lengths | Naive-model |
|---|---|
| $t_1$ | 0.017 (0.002-0.059) |
| $t_2$ | 0.011 (0.009-0.120) |
| $t_3$ | 0.059 (0.010-0.051) |
| $t_4$ | 0.027 (0.005-0.040) |
| $t_5$ | 0.040 (0.002-0.100) |
| $\tilde{t}_6$ | 0.055 (0.007-0.150) |
| $\tilde{t}_7$ | 0.040 (0.002-0.060) |
| $t_8$ | 0.021 (0.003-0.050) |
| $t_9$ | 0.025 (0.004-0.044) |
| $\tilde{t}_{10}$ | 0.010 (0.005-0.034) |
| $t_{11}$ | 0.015 (0.004-0.065) |
| $t_{12}$ | 0.150 (0.060-0.150) |

Table 6.1: Posterior means ($2.5\%$ and $97.5\%$ quantiles) for the branch lengths when using algorithm (4.9). Notice that $\tilde{t}_6 = t_1 + t_6$, $\tilde{t}_7 = t_3 + t_7$, and $\tilde{t}_{10} = t_{10} + \tilde{t}_7 - t_3$.

| | Naive-model |
|---|---|
| Frequencies | |
| $\pi_A$ | 0.28 (0.25-0.36) |
| $\pi_C$ | 0.17 (0.11-0.22) |
| $\pi_G$ | 0.33 (0.25-0.36) |
| $\pi_T$ | 0.22 (0.18-0.25) |
| Rates | |
| $r_{AC}$ | 0.28 (0.19-0.39) |
| $r_{AG}$ | 0.20 (0.15-0.30) |
| $r_{AT}$ | 0.09 (0.02-0.22) |
| $r_{CG}$ | 0.05 (0.01-0.12) |
| $r_{CT}$ | 0.20 (0.10-0.27) |
| $r_{GT}$ | 0.18 (0.10-0.24) |

Table 6.2: Posterior means ($2.5\%$ and $97.5\%$ quantiles) for the nucleotide frequencies and rates of substituion when using algorithm (4.9).

## 6.5 Results using HMM structure

### 6.5.1 Inferring tree topologies

As for the previous approach, algorithm (4.12) with GTR model was run for 600000 iterations with the first 100000 discarded as burn-in. Notice that although the forward-backward sampler requires few iterations and a little burn-in to achieve convergence, we ran the algorithm for a bigger number of replicates as some of the remaining parameters of the model can be more difficult to sample (Ronquist, Huelsenbeck and van der Mark 2005). This algorithm concerns the case where dependencies between neighboring sites of genomical sequences are modelled by using a first order spatial correlation. The results of this procedure are reported in Figure 6-4. It is evident that the performance of the HMM classifier is improved in comparison with the naive; the pattern is quite strong and the classification based on the mean of the posterior probabilities satisfactory. The position of the two regions (one between nucleotides $1-330$ which includes edge $(7,6)$ and the other between $331-350$ which contains edge $(9,10)$) is represented in the figures by vertical dashed lines. The choice of the breakpoint locations is arbitrary, in that the two regions are chosen by looking at the posterior probabilities of **S** for the tree topologies. Of course the breakpoint locations could be changed choosing for example $1-320$ and $321-350$, but the conclusions would still be the same. Looking in detail at Figures 6-4 and 5-13, we spot a similarity; the trend coming out from the two plots reporting the posterior probabilities of **S** is clear but with some spikes. These spikes are present in all the figures reporting the estimated posterior probabilities concerning the $rps11$ dataset. The reason for this is not entirely clear, but may be related to the little DNA site variability of the data. Here, further investigation is needed which is beyond the scope of the present work.

### 6.5.2 Parameter estimates

Tables 6.3 and 6.4 contain the estimates of the twelve branch lengths and parameters of the model of evolution. Comparing the nucleotide frequencies and rates of substitution estimates with those of the naive approach, the results match well, although the naive estimates appear more variable. However, the branch length estimates differ from those of the naive approach. These results confirm our simulation findings. In fact, the estimates of the branch lengths for the naive approach are expected to be worse as the naive classifier does not

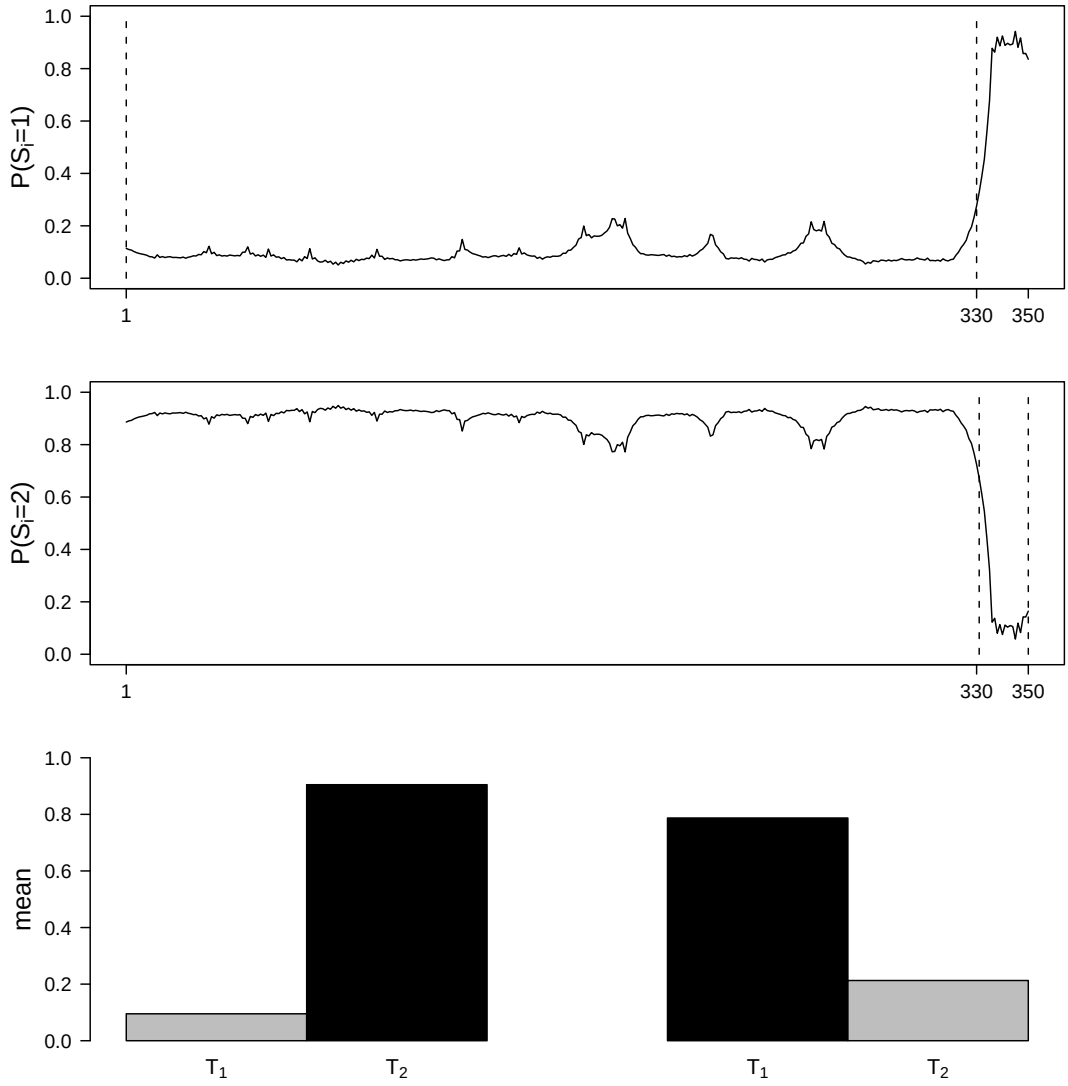



Figure 6-4: Inferred HGTs with HMMs for the ribosomal protein gene $rps11$ data. The first two row plots show the posterior probabilities for the two states. The bar plots (bottom) show the mean of the posterior probabilities of **S** for the two topologies in each region.

allow us to obtain satisfactory tree allocations.

| Edge lengths | HGT-model |
|---|---|
| $t_1$ | 0.026 (0.010-0.041) |
| $t_2$ | 0.044 (0.010-0.075) |
| $t_3$ | 0.020 (0.008-0.031) |
| $t_4$ | 0.015 (0.010-0.020) |
| $t_5$ | 0.026 (0.012-0.040) |
| $\tilde{t}_6$ | 0.082 (0.055-0.110) |
| $\tilde{t}_7$ | 0.070 (0.053-0.090) |
| $t_8$ | 0.004 (0.002-0.006) |
| $t_9$ | 0.035 (0.025-0.043) |
| $\tilde{t}_{10}$ | 0.005 (0.002-0.007) |
| $t_{11}$ | 0.023 (0.013-0.031) |
| $t_{12}$ | 0.105 (0.080-0.125) |

Table 6.3: Posterior means ($2.5\%$ and $97.5\%$ quantiles) for the branch lengths when using algorithm (4.12). Notice that $\tilde{t}_6 = t_1 + t_6$, $\tilde{t}_7 = t_3 + t_7$, and $\tilde{t}_{10} = t_{10} + \tilde{t}_7 - t_3$.

| | HGT-model |
|---|---|
| Frequencies | |
| $\pi_A$ | 0.30 (0.27-0.34) |
| $\pi_C$ | 0.18 (0.16-0.20) |
| $\pi_G$ | 0.30 (0.26-0.34) |
| $\pi_T$ | 0.22 (0.18-0.26) |
| Rates | |
| $r_{AC}$ | 0.28 (0.20-0.38) |
| $r_{AG}$ | 0.23 (0.16-0.28) |
| $r_{AT}$ | 0.10 (0.04-0.19) |
| $r_{CG}$ | 0.05 (0.03-0.07) |
| $r_{CT}$ | 0.18 (0.10-0.25) |
| $r_{GT}$ | 0.16 (0.08-0.25) |
| Prob. of no HGT | |
| $\nu$ | 0.99 (0.97-1.00) |

Table 6.4: Posterior means ($2.5\%$ and $97.5\%$ quantiles) for the nucleotide frequencies, rates of substituion and probability of not changing topology when using algorithm (4.12).

Overall, looking at the autocorrelation function and trace plots of the branch lengths, nucleotide frequencies, rates of substitution and probability of not changing topology in Figures 8-5–8-8 in the Appendix the convergence of the MCMC chains is satisfactory and the acceptance rates of the proposal mechanisms range from $20\%$ to $50\%$.

### 6.5.3 Choice of the model of evolution

From the estimates of the nucleotide frequencies and rates of substitution, the GTR looks a sensible choice for modelling these data. However using a simpler model (for example Jukes Cantor), the results for the posterior probabilities of the tree topologies are not changing significantly (see Figure 6-5). This is in agreement with the simulation results presented in Figure 5-15 of Chapter 5.

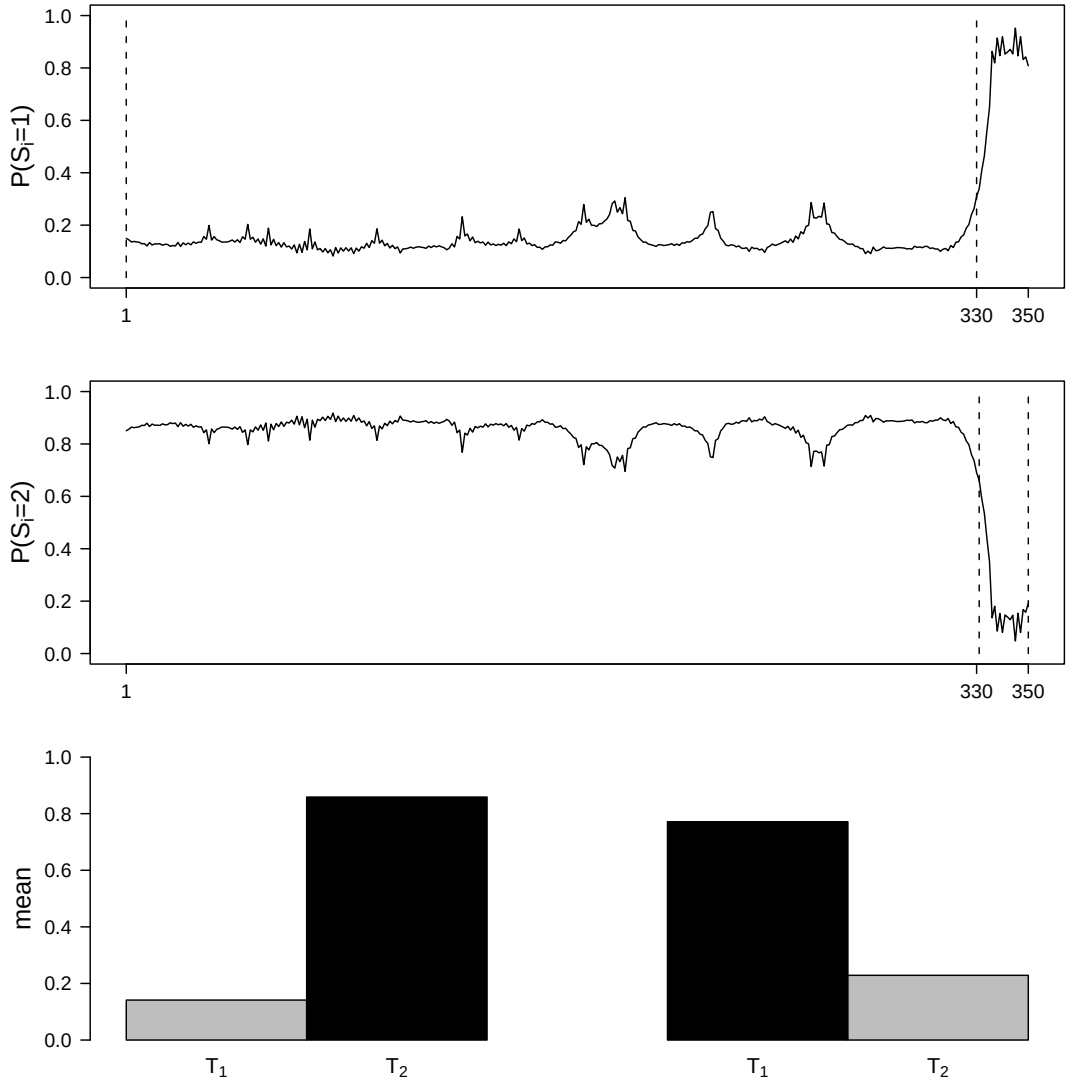



Figure 6-5: Inferred HGTs with Jukes Cantor model for the ribosomal protein gene $rps11$ data. The first two row plots show the posterior probabilities for the two states. The bar plots (bottom) show the mean of the posterior probabilities of **S** for the two topologies in each region.

### 6.5.4 Different types of phylogenetic networks

As already mentioned in Section 6.3 we assume that the species tree and, more importantly, the location and number of reticulation events are known. Assuming that the species tree is available is not completely unreasonable. In fact for many species the organismal tree underlying the network is available or at least can be inferred with high degree of probability or confidence. However, it is less plausible that we know the location and number of all reticulation events. In this section we assess:

1. the effect of choosing different reticulation events from those chosen in Section 6.3 but with the underlying species tree fixed;

2. the effect of choosing more HGTs than those chosen in the previous section.

#### Choosing different reticulation events

Suppose we have a new phylogenetic network (see Figure 6-6). This is different from that in Figure 6-1 as the position of the two horizontal gene transfers is not the same. In fact, in Figure 6-1 Annona is horizontally transferring genetic material to Cabomba, and Tradescantia to Annona, whereas according to Figure 6-6 Platanus is laterally transferring genetic material to Cabomba, and Abelia to Cabomba: the trees in Figure 6-7 are different from those in Figure 6-2. Notice that the trees induced by the network in Figure 6-6 are two (not four). As explained earlier, the amount of DNA available does contain only the regions which include HGTs, hence the species tree is not present. Also, since the 'head' of the two HGTs is in the same tree branch, it is impossible that the two HGT events occur together, meaning that the tree containing both the reticulation events is not included.

The case described here is similar to misspecification 3 in Section 5.4.5. We ran algorithm (4.12) for 15000 iterations with the first 100 discarded as burn-in. The results reported in Figure 6-8 clearly exhibit noisy probabilities with some spikes for the two tree topologies. Importantly, they are centered about 0.5, hence not preferring any of the two trees for all the alignment. This suggests that the HGTs in Figure 6-1 are better supported by the data than the HGTs in Figure 6-6.

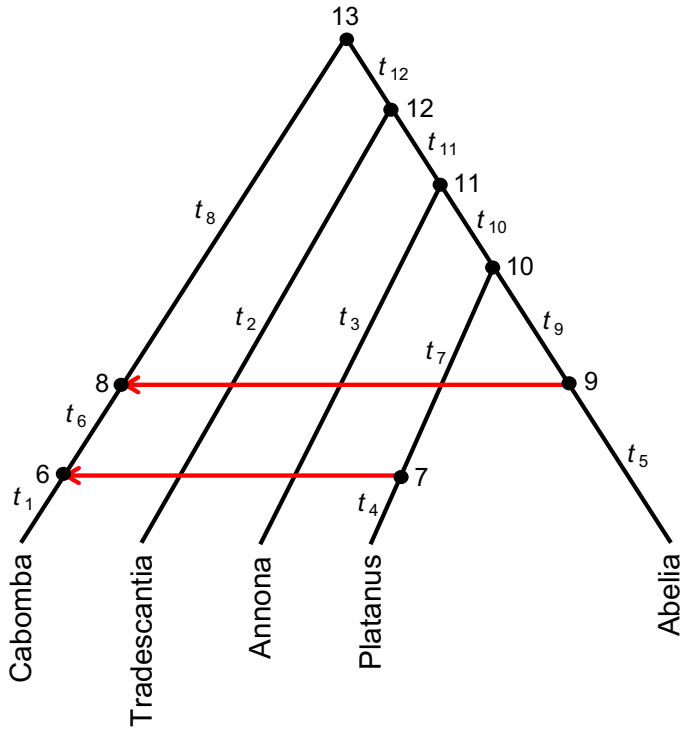
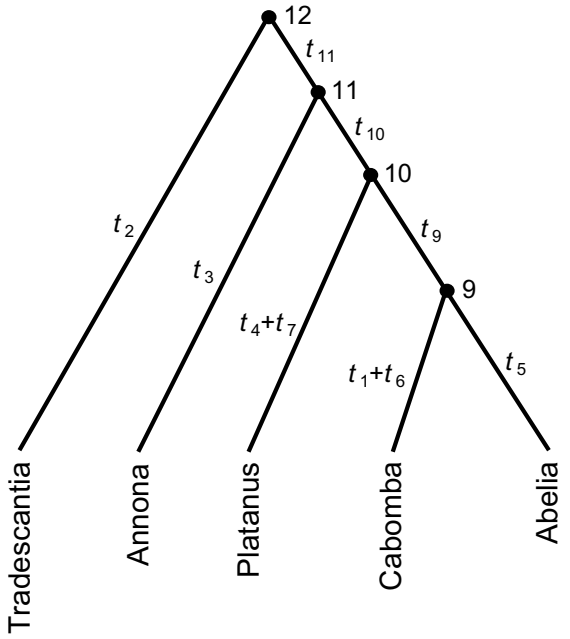
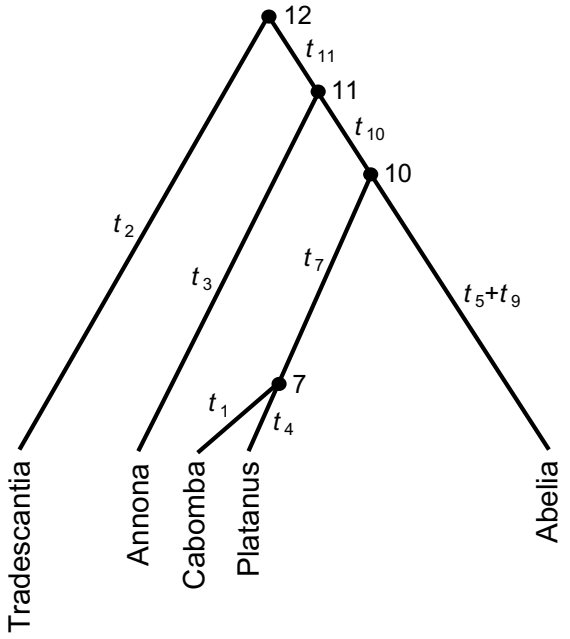



Figure 6-6: An alternative phylogenetic network of the ribosomal protein gene $rps11$ data with $R = 2$ reticulation events: (7,6) and (9,8).




(a) The horizontally transferred gene tree $T_1$ including the edge (9,8).




(b) The horizontally transferred gene tree $T_2$ including the edge (7,6).

Figure 6-7: Trees induced by the network in Figure 6-6.

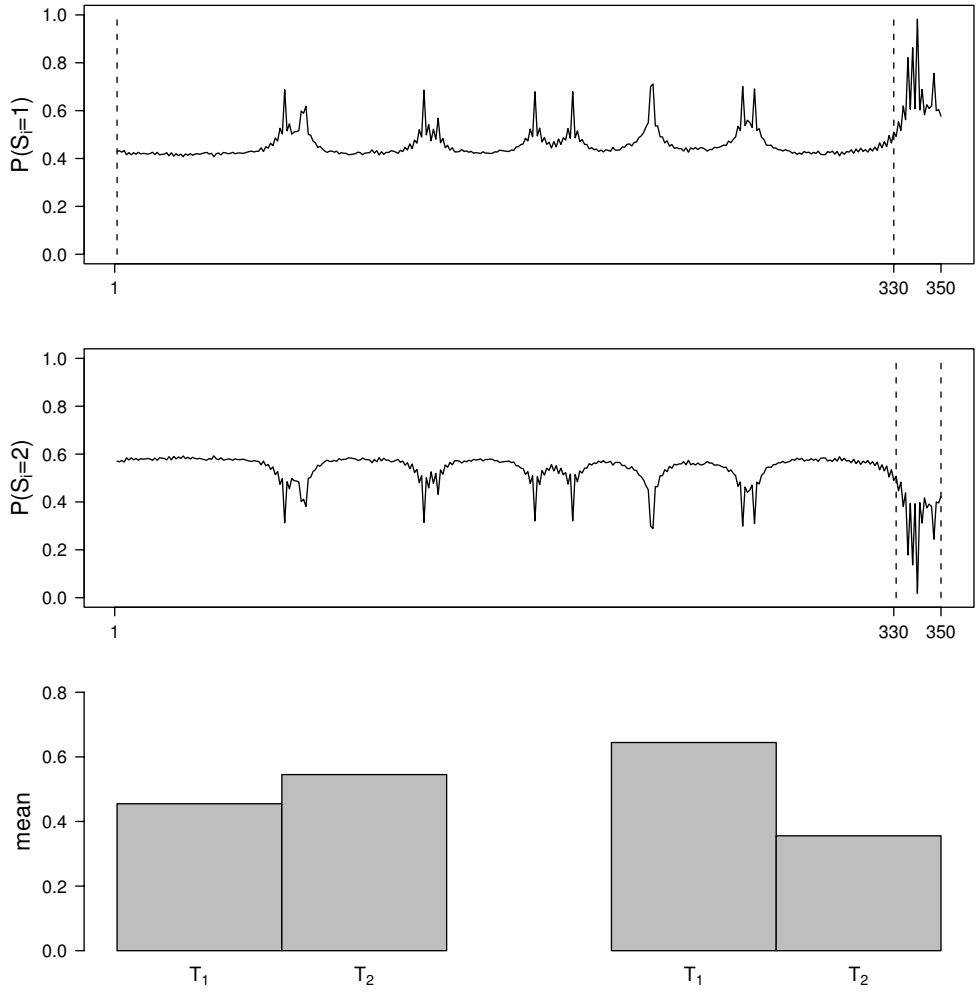



Figure 6-8: Posterior probabilities for two tree topologies (first two top rows) and bar plots (bottom) showing the classification on the ribosomal protein gene $rps11$ with different HGTs.

**Choosing more reticulation events**

Consider the phylogenetic network presented in Figure 6-9, with its constituent trees depicted in Figure 6-10. Comparing this network with that in Figure 6-1, the only difference is in the additional reticulation, represented by the horizontal transfer of genetic material form Tradescantia to Platanus. The trees induced by the network in Figure 6-10 are two and not $2^R = 8$, for the same reasons discussed above and in Section 6.3 and because the first two trees are exactly the same as those reported in Figure 6-2 (but with labels changed accordingly). Notice that here we chose the subset of trees looking at previous works (Bergthorsson *et al*., 2003; Snir and Tuller, 2009). However, as already mentioned, a heuristic method could also be used.

The case considered here is very close to that of misspecification 2 in Section 5.4.5. Algorithm (4.12) was run for 15000 iterations with the first 100 discarded as burn-in. The results reported in Figure 6-11 show that between nucleotides $i = 1 - 330$ tree $T_2$ is always preferred to any other, whereas between $i = 331 - 350$ tree $T_1$ has the highest posterior probability. So, by using a network containing more HGTs, the results are not significantly different from those when using the network in Figure 6-1, indicating once again, that the data better support the HGTs in Figure 6-1 rather than those implied by the other networks. However, it is worth noting that several different types of network can be obtained by trying all possible combinations of reticulation events. This can be a challenging task as computational cost can become burdensome.

## 6.6 Discussion

An application of the Bayesian phylogenetic network algorithm to the ribosomal protein gene $rps11$ data has been presented. In 2003 Bergthorsson *et al*. provided the first unambiguous evidence that this dataset underwent HGTs. On the basis of this work and others (e.g. Bergthorsson *et al*., 2004 and Snir and Tuller, 2009) all the parameters of the phylogenetic network have been inferred. The evidence of our study is that some HGTs seem more likely to have taken place than others. Of course we made the simplistic assumption that we know *a priori* where the HGTs are occurring. While this can be true for some species, the same cannot be said for others. Great care has to be taken when dealing with biological organisms which underwent reticulation events, and a more flexible and general algorithm is needed in order to select a set of HGTs, and hence a set of trees, from which the DNA sequences are thought to be

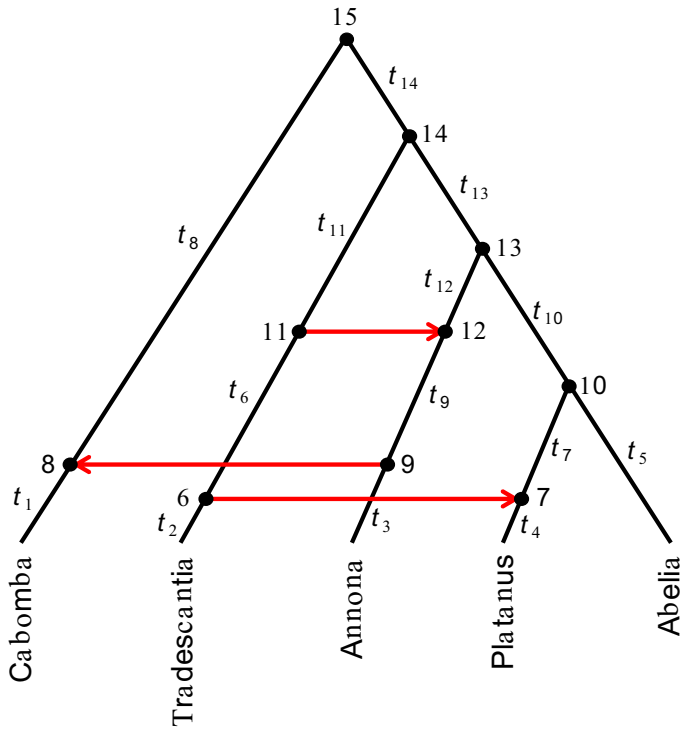
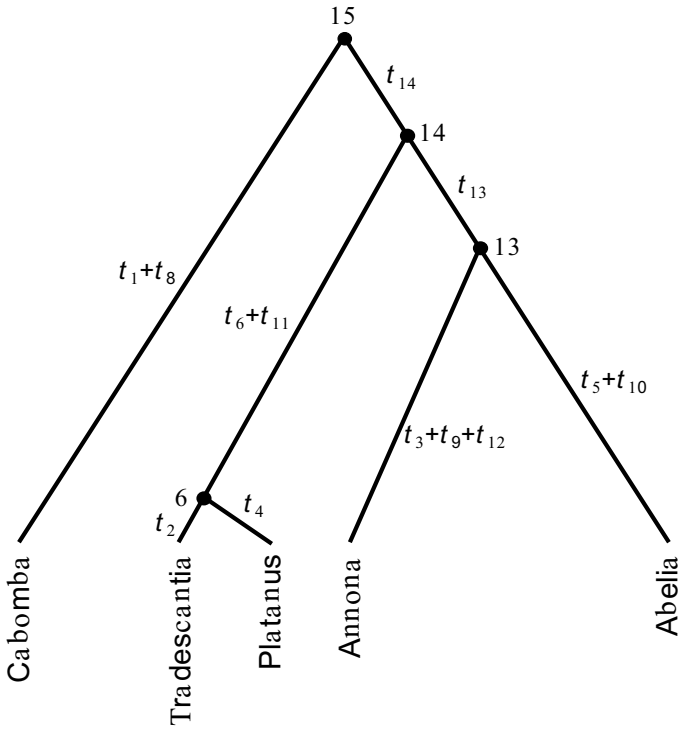
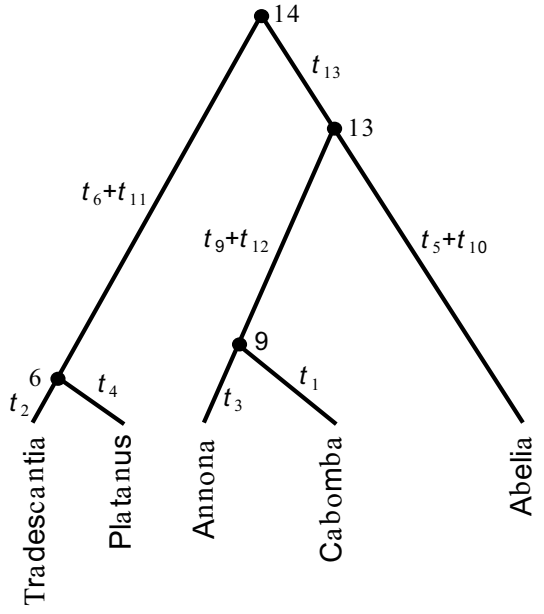



Figure 6-9: Phylogenetic network of the ribosomal protein gene $rps11$ data with $R = 3$ reticulation events: (6,7), (9,8) and (11,12).




(a) The horizontally transferred gene tree $T_3$ including the edge (6,7).




(b) The horizontally transferred gene tree $T_4$ including the edges (6,7) (9,8).

Figure 6-10: Trees induced by the network in Figure 6-9.

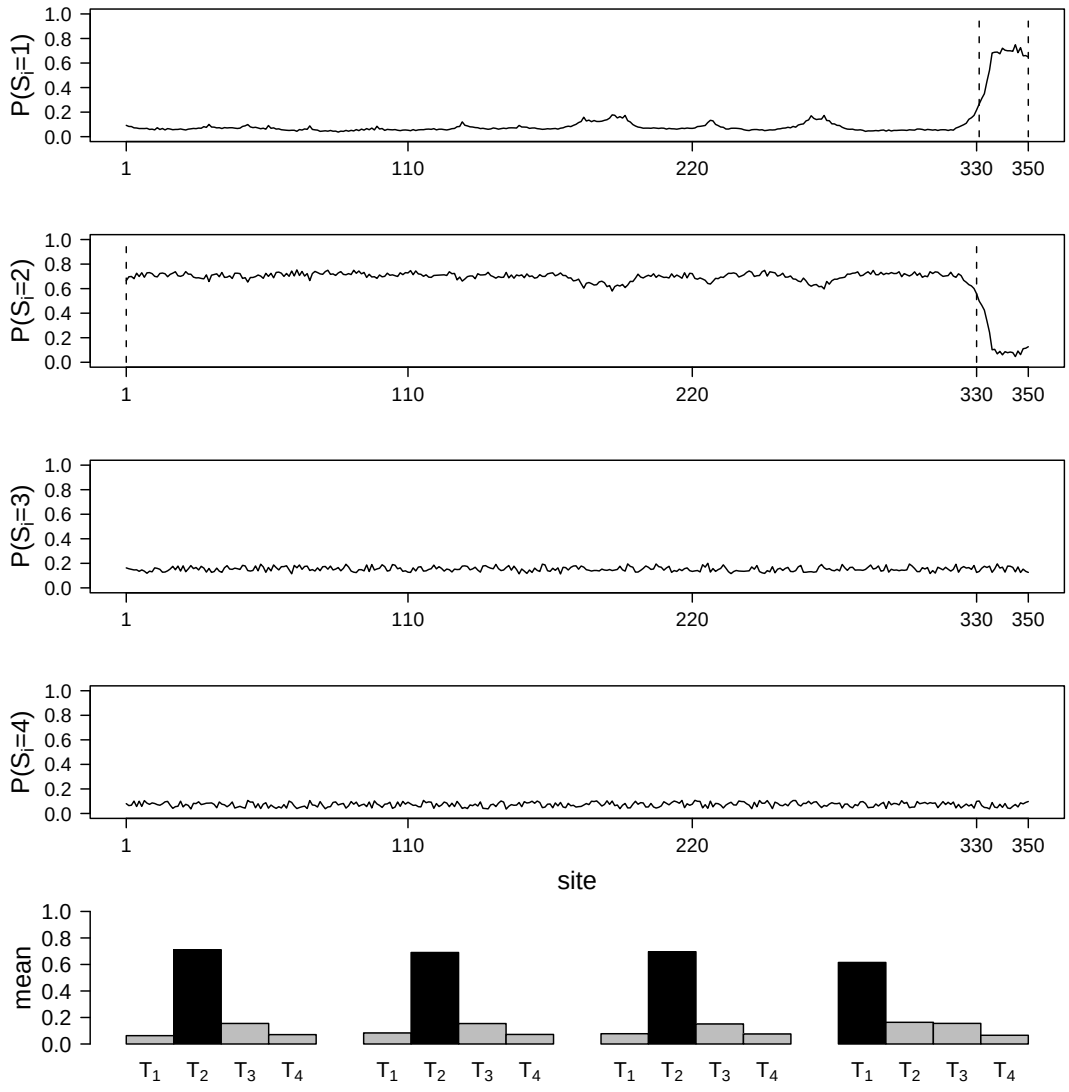



Figure 6-11: Posterior probabilities for four tree topologies (first four top rows) and bar plots (bottom) showing the classification on the ribosomal protein gene $rps11$ with more HGTs.

evolved. This issue will be addressed in the next chapter where an improved and more general algorithm will be proposed.

# Chapter 7

# Stochastic search variable selection for identifying tree topologies

## 7.1 Introduction

In this chapter we propose a Bayesian method for identifying tree topologies, and hence reticulation events at the species level (HGT and HS) in multiple DNA sequences. As alluded to previously, the Bayesian method developed in Chapter 4 is not sufficiently flexible. In fact, for cases where HGT and HS events are many, the tree topologies are not easily enumerated, and a more flexible and general algorithm is needed in order to avoid exploring the entire space of tree topologies. For this reason, we need to restrict the set of reticulations (or of tree topologies) since many of them would contribute little to the likelihood of the data. This can be achieved by using a sampling scheme based on a variable selection method called stochastic search variable selection (SSVS). We firstly describe the concept of SSVS in regression models, and then adapt it to our framework by turning the problem from a variable selection setting into a tree topology selection setting. Finally, we show the performance of the algorithm on simulated data, and apply it to the ribosomal protein gene $rps11$ data.

## 7.2 Stochastic search variable selection

In variable selection problems, the list of models under consideration corresponds to the $2^K$ possible subsets of a set of $K$ candidate covariates. Clearly, the number of models rapidly becomes enormous as $K$ increases, so there is a need for efficient methods in the search for high posterior probability mod-

els. A strategy for addressing this problem is to use SSVS which was first introduced and developed by George and McCulloch (1993) in the context of linear regression models. SSVS is a procedure to select promising subsets of predictor variables in the defined design matrix. This procedure is based on embedding the entire regression setup in a hierarchical Bayes normal mixture model, where latent variables are used to specify choices of subsets. The idea behind this method is that statistical models can be represented by a set of binary latent indicator variables $\boldsymbol{\gamma} = (\gamma_1, \ldots, \gamma_K)$, where $\gamma_k = 1$ or $0$ represents the presence or absence of covariate $k$ in the model, respectively. So only a set of covariates dictated by the data is maintained. Variables with little probabilistic support under the data are removed from inference avoiding the overwhelming computational burden. Specifically, the canonical regression setup and prior distributions of the model parameters are as follows:

$$\begin{aligned}
\mathbf{Y}|\boldsymbol{\beta}, \sigma^2 &\sim N_n\left(\mathbf{X}\boldsymbol{\beta}, \sigma^2\mathbf{I}\right) \\
\beta_k|\gamma_k &\sim (1-\gamma_k)\, N\left(0, \tau^2\right) + \gamma_k N\left(0, c^2\tau^2\right),\ k = 1, \ldots, K \\
\sigma^2|\boldsymbol{\gamma} &\sim Inv - Gamma(\nu_\gamma/2, \nu_\gamma\lambda_\gamma/2) \\
\gamma_k &\sim Ber(0.5),\ k = 1, \ldots, K.
\end{aligned}$$

Here, $\mathbf{Y}$ is a vector of response measurements, $\mathbf{X}$ is an $n \times K$ matrix which contains $K$ predictors and $n$ observations. $\beta$ is a mixture prior, with parameters $\tau^2$ and $c^2$ chosen so that $\tau^2$ is small and $c^2\tau^2$ is large. If $\gamma_k = 0$, then the magnitude of the effect $\beta_k$ is small and the prior distribution for $\beta_k$ forces this parameter to be close to zero. If $\gamma_k = 1$, then the magnitude of the effect $\beta_k$ is large and a nonzero estimate of $\beta_k$ should be included in the model and its posterior distribution will largely be determined by the data. $\nu_\gamma$ and $\lambda_\gamma$ are prior parameters which may depend on $\boldsymbol{\gamma}$ to incorporate the dependence between $\boldsymbol{\beta}$ and $\sigma^2$. On the basis of the prior specifications described above a Gibbs sampler can be used to generate samples from the posterior distribution on the set of possible covariate subsets. Those subsets with higher probability can be identified by their more frequent appearance in the Gibbs sampler.

SSVS has not been restricted to the linear regression context. In fact it has been adopted for more sophisticated models such as generalized linear (George *et al.*, 1996), and additive models (Reich *et al.*, 2009). SSVS has also recently been applied to complex genetic modelling such as multiple quantitative trait loci (Yi *et al.*, 2003), genome selection (Verbyla *et al.*, 2009), and recombinant phylogenetic models (Webb *et al.*, 2009).

## 7.3 Stochastic search tree selection

A problem with the approach proposed in Chapter 4 is that the state space of the tree topologies can be vast for all the possible combinations of reticulation events which can occur. To overcome this issue we adapt SSVS to our context of phylogenetic networks by providing the stochastic search tree selection (SSTV) algorithm. In this stochastic selection approach only a random subset of topologies supported by the data is maintained. The idea behind this approach is similar to that previously described, where statistical models can be represented by a set of binary indicator variables $\boldsymbol{\gamma} = (\gamma_1, \ldots, \gamma_K)$, where $\gamma_k = 1$ or $0$ represents the presence or absence of topology (reticulations) $k$ in the model, respectively. Topologies with little probabilistic support under the data are removed from the inference leading to massive computational savings.

In order to infer the topologies and the other parameters, we employ a Markov chain Monte Carlo approach to find the posterior probability of topology $S_i$ for each site in the data, and the evolution model parameters. In particular, the $(j+1)^{th}$ sample for all the parameters is obtained as shown in Chapter 4 but with the difference that before sampling the $S_i$ we first allow the topology HMM to update, and then we sample a new path given the current states using the stochastic forward-backward algorithm.

### 7.3.1 Methodology

Each time we update the topology HMM we may:

(a) add a new tree topology (birth step), i.e. select a tree topology such that $\gamma_k = 0$ and propose to set $\gamma_k = 1$;

(b) delete an existing tree topology (death step), i.e. select a tree topology with $\gamma_k = 1$ and set to $\gamma_k = 0$;

(c) make a rearrangement of an existing tree topology.

The new tree topology to add in step (a) is found by uniformly choosing one of the topologies which is not included in the current model. The proposed topology to delete in step (b) is simply chosen uniformly from the trees that are currently in the model. Although theoretically we can produce samples from the correct posterior using just birth and death moves this is often not the best strategy. Combining moves (a) and (b) with move (c) have been shown to work

well (see, for example, Denison *et al*., 1998). Hence we incorporate into the algorithm step (c), which just resamples the tree topology. This step proposes to alter a randomly chosen topology by swapping a tree topology chosen from the current trees for a randomly chosen topology not included in the current model. In order to jump between states with different tree topologies we need a prior on the number of topologies $\sum_k \gamma_k$. A natural choice for this is a Poisson distribution (with parameter $\lambda$). Let $k_T$ be the current number of topologies in the model, $k_T = \sum_k \gamma_k$, then

$$P(k_T) = \frac{\lambda^{k_T} \exp(-\lambda)}{k_T!}. \tag{7.1}$$

In practice, a Poisson distribution truncated to $k_T < k_{\max}$, for a suitable choice of $k_{\max}$, is adopted. In this case $k_{\max} = K$, where $K$ is total number if trees contained in a network. The $\lambda$ is chosen *a priori* as the expected number of tree topologies in the data and, as in Webb *et al*. (2009), is set $\lambda = 5$. Viallefont *et al*. (2002) show that in variable dimension problems, the use of a truncated Poisson may lead to model sensitivity. For these reason is important to check the robustness of the conclusions for different values of $\lambda$ (see next section). The moves described above occur with probabilities:

$$\begin{cases} b_{k_T} = c \min\left\{1, P\left(k_{T+1}\right)/P\left(k_T\right)\right\} \\ d_{k_T} = c \min\left\{1, P\left(k_{T-1}\right)/P\left(k_T\right)\right\} \\ m_{k_T} = 1 - b_{k_T} - d_{k_T} \end{cases} \tag{7.2}$$

where the tuning constant $c$ controls the rate at which move types which change dimension are proposed. Here we take $c = 0.4$ but other values are equally valid, provided that $c \in [0, \frac{1}{2}]$ as, if $c > \frac{1}{2}$, then the sum of the probabilities $b_{k_T}$ and $d_{k_T}$ could be greater than 1 for same values of $k_T$. The probabilities in (7.2) are as in Webb *et al*. (2009) and satisfy the detailed balance condition

$$b_{k_T} P(k_T) = d_{k_{T+1}} P(k_{T+1}).$$

Using the notation of Green (1995), the acceptance probability for each move types in our problem is

$$\alpha = \min\left(1, \text{likelihood ratio} \times \text{prior ratio} \times \text{proposal ratio} \times \text{Jacobian.}\right) \tag{7.3}$$

For the move step (c) the prior ratio and the proposal ratio are both 1 since all collections of the same number of topologies have the same prior probabil-

ity and the proposals are made from the same distribution.

For the birth step (a) the prior ratio is given by

$$\text{prior ratio} = \frac{P(k_{T+1})}{P(k_T)} \tag{7.4}$$

where $P(k_{T+1})$ and $P(k_T)$ are the priors given in (7.1) and the second term assumes that each possible dimension $k_T \in 1, \ldots, K$ is equally likely. The corresponding proposal ratio is given by

$$\text{proposal ratio} = \frac{d_{k_{T+1}}}{k_{T+1}} \frac{K - k_T}{b_{k_T}}. \tag{7.5}$$

The proposal ratio involves understanding both the birth and the converse death step. When adding a new tree topology we use the proposal density $b_{k_T}/(K - k_T)$. This is made up of the probability of actually attempting the birth step in equation (7.2) together with that of choosing a particular new tree topology. This can be done in $K - k_T$ ways as the new tree topology must be distinct to the $k_T$ topologies and there are only $K$ possibilities in total. The probability of proposing the reverse move is equal to $d_{k_{T+1}}/(k_{T+1})$. This is just the probability of proposing a death step and then choosing the proposed tree topology as the one to remove. Since we do not propose parameters that change across dimension, the Jacobian in (7.3) is equal to one.

So, it follows from equations (7.4) and (7.5) that the acceptance probability (7.3) for a birth step is

$$\alpha = \min \left\{ 1, \text{likelihood ratio} \times \frac{K - k_T}{k_{T+1}} \right\}$$

and for the death step is the same except that the fraction is inverted. Notice that for simplicity we assumed that the (true) values of the branch lengths are known. This is a strong assumption and its robustness is assessed in the next section.

### 7.3.2 Simulation study

In this section we investigate the performance of the SSTS algorithm on synthetic data. DNA sequences, 600 bases long, are generated. The first 300 bases are evolved along the underlying species tree shown in Figure 2-3a, and the remaining are evolved along the tree that includes the transfer of genetic material from taxon 2 to taxon 4. The data are simulated according to the model of

evolution described in Section 5.2, but with branch lengths fixed to their true values.

**Estimation results**

The parameters of interest are estimated by using the algorithm described in Section 7.3. In particular, we considered eleven possible trees (that is the trees that are consistent with all combinations of reasonable reticulation events), including the two trees from which the data have been generated. These eleven trees have been chosen at random. However a more objective way to select the trees would be to use a more quick and dirty method, such as those described for the split networks. The SSTS algorithm was run for 25000 iterations with the first 5000 discarded as burn-in. Because the aim of this chapter is to select the model containing the right tree topologies, we do not report the estimates for the model parameter (nucleotide frequencies and rates of substitution) since the results are perfectly in line with those obtained in Chapter 5.

| Model | Frequency | Model | Frequency |
|---|---|---|---|
| $M_{1,3}$ | 0.004 | $M_{1,2,5,10}$ | 0.001 |
| $M_{1,5}$ | **0.700** | $M_{1,2,5,11}$ | 0.001 |
| $M_{1,2,3}$ | 0.001 | $M_{1,3,4,5}$ | 0.006 |
| $M_{1,2,5}$ | 0.090 | $M_{1,3,5,6}$ | 0.008 |
| $M_{1,3,5}$ | 0.100 | $M_{1,3,5,7}$ | 0.002 |
| $M_{1,3,11}$ | 0.001 | $M_{1,3,5,8}$ | 0.002 |
| $M_{1,4,5}$ | 0.010 | $M_{1,3,5,9}$ | 0.001 |
| $M_{1,5,6}$ | 0.003 | $M_{1,3,5,10}$ | 0.002 |
| $M_{1,5,7}$ | 0.002 | $M_{1,3,5,11}$ | 0.001 |
| $M_{1,5,8}$ | 0.002 | $M_{1,4,5,10}$ | 0.001 |
| $M_{1,5,9}$ | 0.003 | $M_{1,2,3,4,5}$ | 0.002 |
| $M_{1,5,10}$ | 0.003 | $M_{1,2,3,5,7}$ | 0.001 |
| $M_{1,5,11}$ | 0.005 | $M_{1,2,3,5,9}$ | 0.001 |
| $M_{1,2,3,5}$ | 0.043 | $M_{1,2,3,5,11}$ | 0.001 |
| $M_{1,2,5,6}$ | 0.001 | $M_{1,2,3,5,9,10}$ | 0.001 |
| $M_{1,2,5,8}$ | 0.001 | - | - |

Table 7.1: The frequencies indicate the proportion of times the SSTS algorithm has visited the models.

Table 7.1 reports the proportion of times the models (states) have been visited by the SSTS algorithm. By model we mean here a particular combination of trees, hence of reticulations, from which the data could potentially evolve.

For example model $M_{1,5,11}$ indicates that some sites evolved under tree $T_1$, others under tree $T_5$, and the remaining under tree $T_{11}$. The state that has been visited by the algorithm the most number of times is $M_{1,5}$ (see Figure 7-1). This means that the model with trees $T_1$ and $T_5$ (which are the two trees from which the data have been generated) is better supported by the data than any other model.

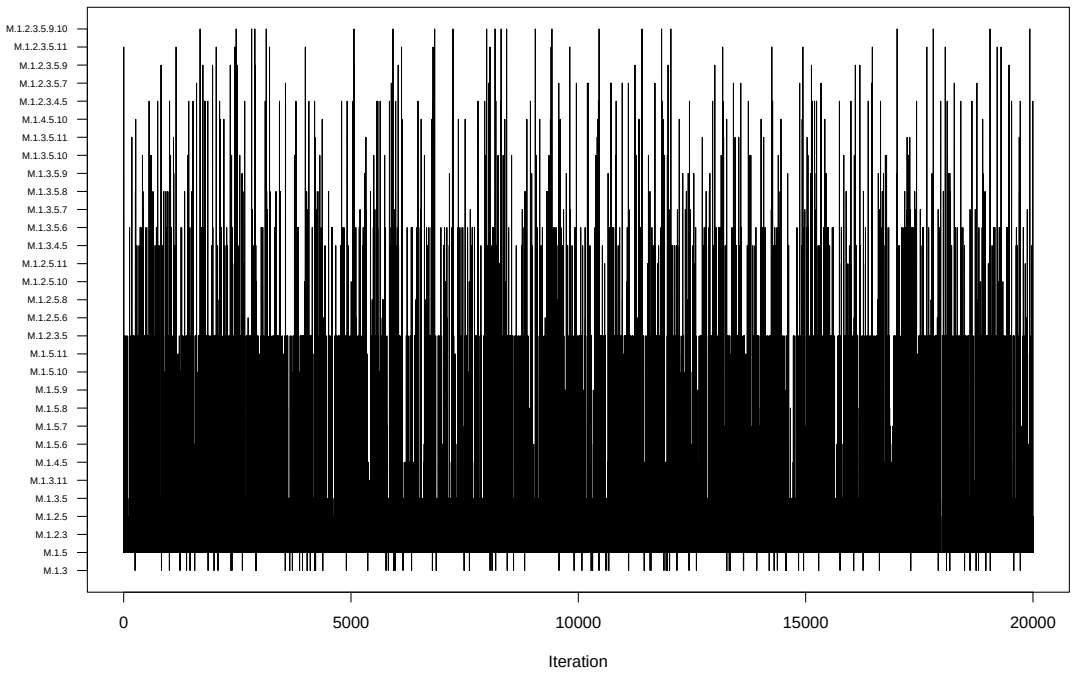



Figure 7-1: Traceplot of the HMM states. The plot shows which models (states) the algorithm has visited. Notice, that $M_{1,5}$ is the model visited by the algorithm the most number of times.

Figures 7-2 and 7-3 show the posterior probabilities for the two topologies, when using the SSTS algorithm combined with the forward-backward procedure, and for the eleven topologies, when using the algorithm described in Section 4.4, respectively, against the sites. Specifically, the results for the case with two tree topologies show a strong signal and an accurate change point estimate. The results for the case with all trees show a clear pattern although some noise is present between sites $300 - 400$. The good performance of the latter procedure comes as no surprise as we could already see in Section 5.4.5 where the MCMC algorithm could select the right tree topologies. However, in this case there are more parameters to estimate and this might cause problems of accuracy, efficiency and computational speed whereas in the SSTS algorithm

only the trees which receive support from the data are selected allowing us to estimate the parameters with more precision and accuracy. To assess the robustness of these findings, additional analyses have been conducted to test for problems of model sensitivity. Specifically the following values of $\lambda$ have been chosen: $1$, $3$, $7$ and $10$. The results (not reported here) show that by changing the value of $\lambda$ the proportions of times the SSTS algorithm has visited the models change, although the final results reach the same conclusions as those for $\lambda = 5$.

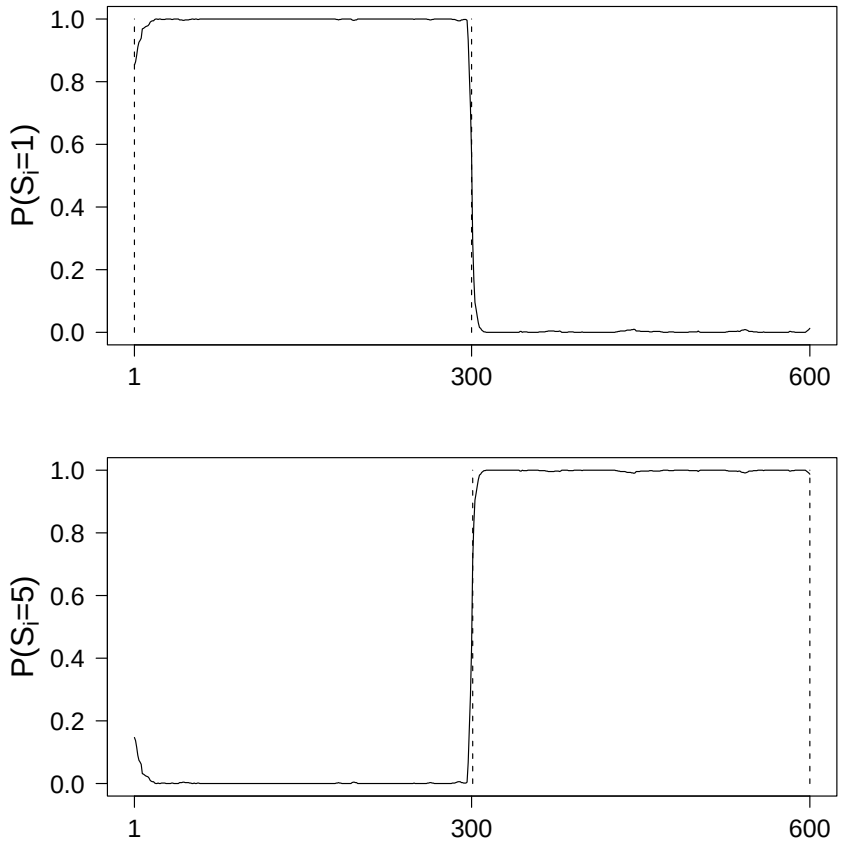



Figure 7-2: HGTs with SSTS. The two plots show the posterior probabilities ($P(S_i = k)$, $k = 1, 5$) for the two tree topologies $T_1$ and $T_5$.

**The effect of wrong branch lengths on SSTS**

For simplification in the simulation setting, the branch lengths have been fixed to their true values. This can be justified by the fact that we are mainly interested in tree selection. However, given that in a real data setting the branch lengths are not known, it is interesting to assess the robustness of the SSTS algorithm with respect to the choice of wrong branch lengths.

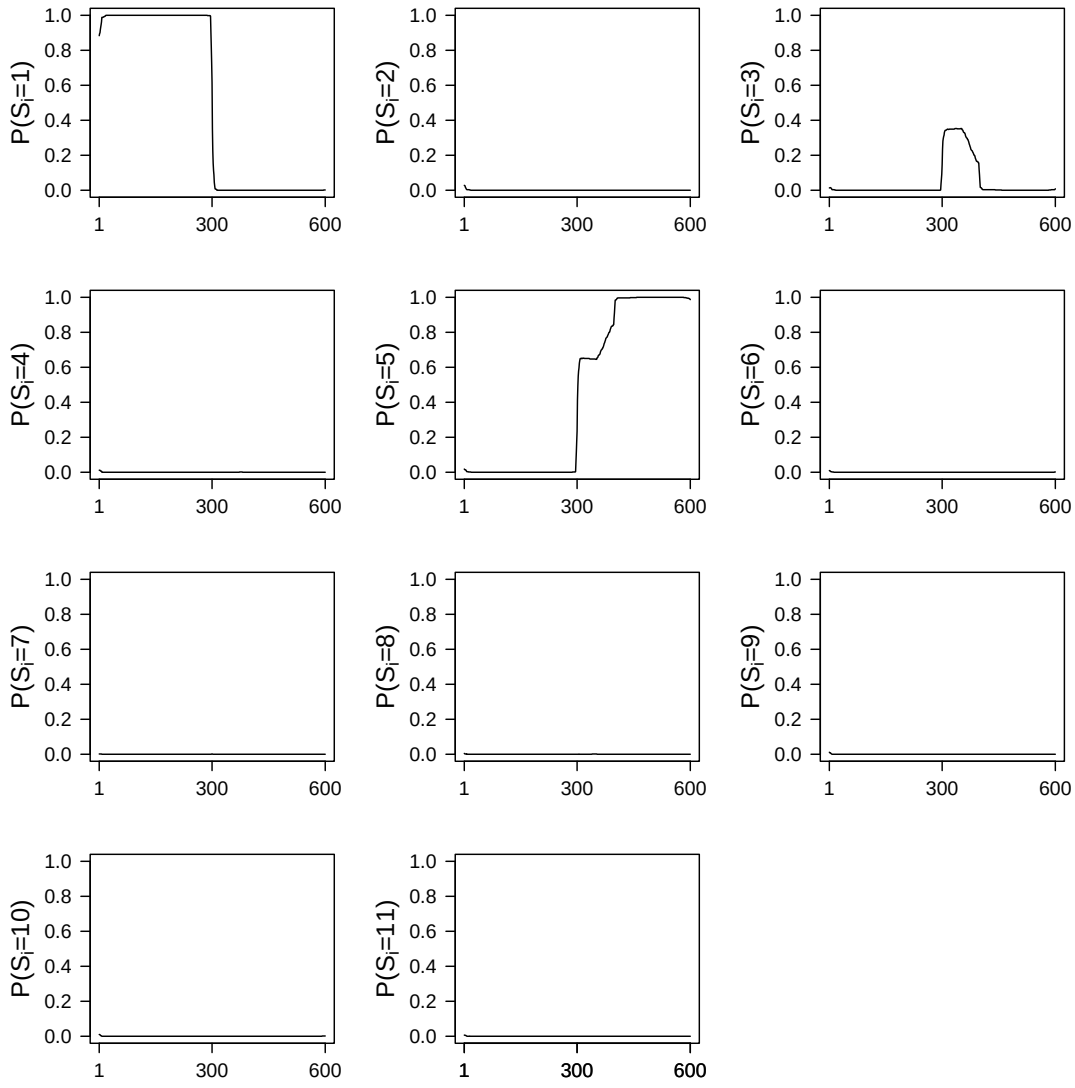



Figure 7-3: HGTs without SSTS. The plots show the posterior probabilities ($P(S_i = k)$, $k = 1, \ldots, 11$) for the eleven tree topologies $T_1$-$T_{11}$.

Table 7.2 reports the proportion of times the models have been visited by the SSTS algorithm with wrong branch length values chosen from a $U[0,1]$, and satisfying the summation restrictions as described in the previous chapters. The state that has been visited the most number of times is $M_{1,5}$. This means that the model with trees $T_1$ and $T_5$ is still better supported by the data, showing that this simplification could be acceptable and reasonable. However, it is important to notice that other models, such as $M_{1,3}$ and $M_{1,3,5}$, have non-negligible support. Model $M_{1,3,5}$ contains two of the three true trees $T_1$ and $T_5$, whereas $M_{1,3}$ includes just one. The high frequencies of these two models can be justified by using the same argument of the previous two chapters. In fact, the two trees $T_3$ and $T_5$ are similar in their structure, and hence more confounded and less distinguishable, given the wrong branch length assignment. In other words the two trees can distort the results in the same way as highly collinear covariates can cause regression parameters to be inefficient or make them quite unstable. Comparing the results of Table 7.2 to those of Table 7.1 two points are worth emphasising; 1) the number of times the algorithm with wrong branch lengths visits the right state is lower as compared to the case where the branch lengths are set to the right values, although the conclusions do not change; 2) there is more variability, hence a loss of efficiency when using the procedure with wrong branch length values.

Figures 7-4 and 7-5 show the posterior probabilities for the two (when using the SSTS algorithm combined with the forward-backward procedure) and eleven topologies (when using algorithm described in Section 4.4), respectively, against the sites with wrong branch length values. The results with two tree topologies show a clear pattern and a good tree allocation, whereas those with eleven topologies allocate half of the sites to the wrong tree; the posterior probability of $T_3$ for sites $300-600$ is systematically higher than that of $T_5$. This means that wrong branch length choices can have a detrimental impact on tree allocations if we use the algorithm of Section 4.4, whereas the SSTS procedure seems to be more robust to the choice of wrong branch length values. By looking in more depth at Figure 7-4, and comparing it to Figure 7-2, it is clear that the results with the wrong branch length values are slightly worse in terms of estimated change point as compared to those with true branch length values, but are still reliable and accurate.

| Model | Frequency | Model | Frequency |
|---|---|---|---|
| $M_{1,3}$ | **0.135** | $M_{1,3,5,9}$ | 0.012 |
| $M_{1,5}$ | **0.452** | $M_{1,3,5,10}$ | 0.014 |
| $M_{1,7}$ | 0.001 | $M_{1,3,5,11}$ | 0.002 |
| $M_{1,11}$ | 0.001 | $M_{1,3,7,9}$ | 0.003 |
| $M_{1,2,3}$ | 0.010 | $M_{1,3,7,10}$ | 0.005 |
| $M_{1,2,5}$ | 0.001 | $M_{1,3,5,11}$ | 0.001 |
| $M_{1,3,4}$ | 0.001 | $M_{1,3,9,10}$ | 0.001 |
| $M_{1,3,5}$ | **0.160** | $M_{1,3,9,11}$ | 0.002 |
| $M_{1,3,6}$ | 0.003 | $M_{1,5,7,9}$ | 0.004 |
| $M_{1,3,7}$ | 0.009 | $M_{1,5,7,10}$ | 0.003 |
| $M_{1,3,9}$ | 0.025 | $M_{1,5,7,11}$ | 0.002 |
| $M_{1,3,10}$ | 0.031 | $M_{1,5,9,11}$ | 0.001 |
| $M_{1,3,11}$ | 0.002 | $M_{1,5,10,11}$ | 0.001 |
| $M_{1,4,5}$ | 0.001 | $M_{1,9,10,11}$ | 0.001 |
| $M_{1,5,6}$ | 0.001 | $M_{1,2,3,7,9}$ | 0.001 |
| $M_{1,5,7}$ | 0.050 | $M_{1,3,4,5,7}$ | 0.002 |
| $M_{1,5,10}$ | 0.006 | $M_{1,3,4,5,9}$ | 0.001 |
| $M_{1,5,11}$ | 0.007 | $M_{1,3,4,9,10}$ | 0.001 |
| $M_{1,7,9}$ | 0.001 | $M_{1,3,5,7,9}$ | 0.004 |
| $M_{1,2,3,5}$ | 0.002 | $M_{1,3,5,7,10}$ | 0.002 |
| $M_{1,2,3,7}$ | 0.002 | $M_{1,3,5,8,10}$ | 0.001 |
| $M_{1,2,3,9}$ | 0.002 | $M_{1,3,5,9,10}$ | 0.001 |
| $M_{1,3,4,5}$ | 0.002 | $M_{1,3,5,9,11}$ | 0.001 |
| $M_{1,3,4,10}$ | 0.001 | $M_{1,3,5,10,11}$ | 0.001 |
| $M_{1,3,5,6}$ | 0.002 | $M_{1,3,7,10,11}$ | 0.001 |
| $M_{1,3,5,7}$ | 0.022 | $M_{1,2,3,5,7,9}$ | 0.001 |
| $M_{1,3,5,8}$ | 0.001 | - | - |

Table 7.2: The frequencies indicate the proportion of times the SSTS algorithm with wrong branch length values has visited the models.

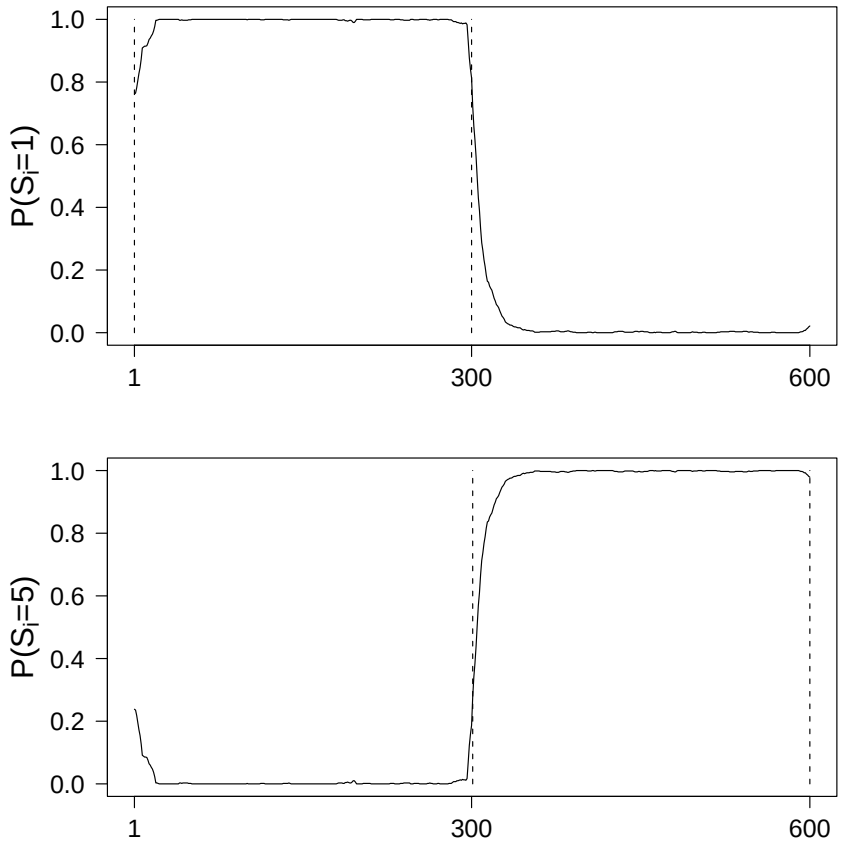



Figure 7-4: HGTs with SSTS and wrong branch lengths. The two plots show the posterior probabilities ($P(S_i = k)$, $k = 1, 5$) for the two tree topologies $T_1$ and $T_5$ when using the SSTS algorithm with wrong branch length values.

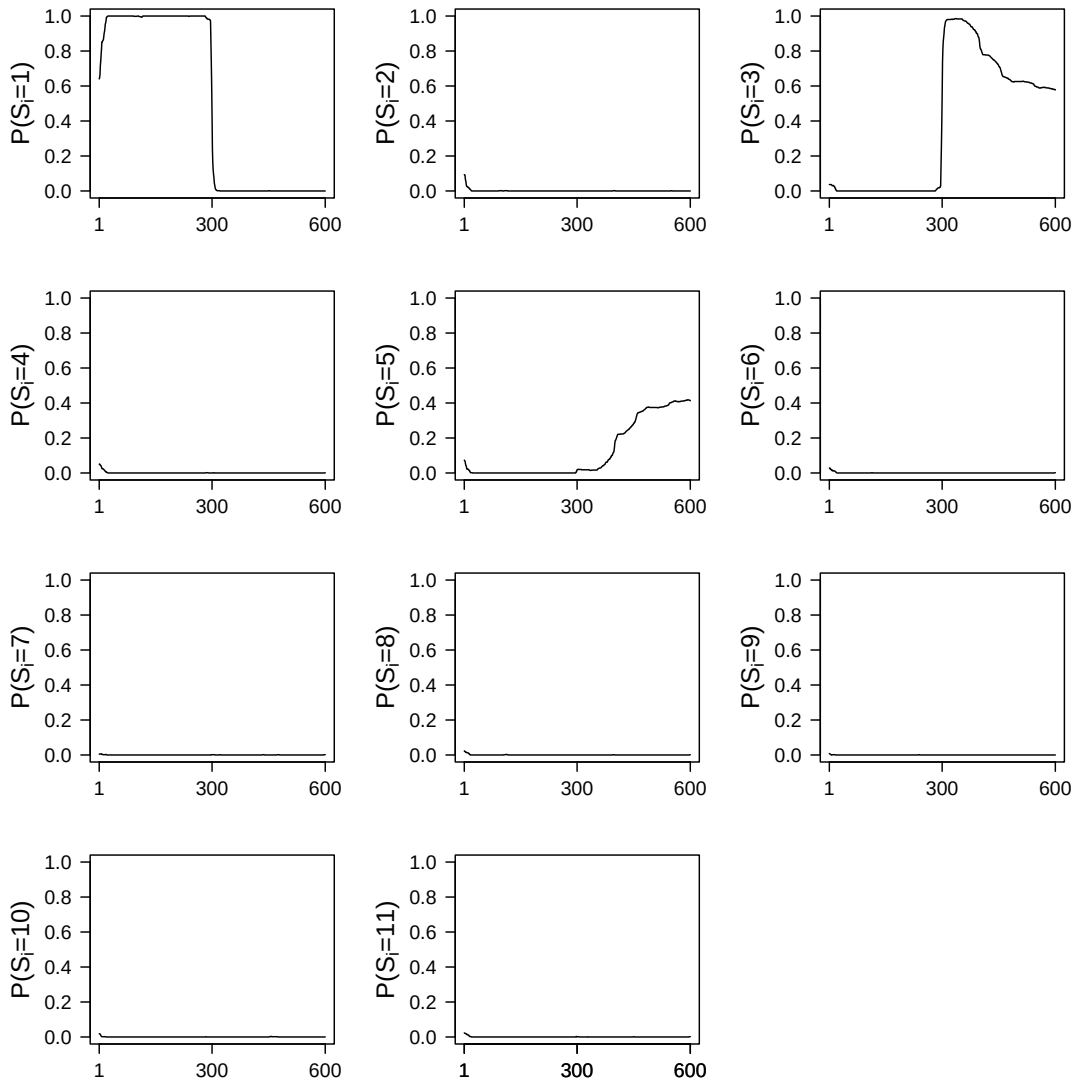



Figure 7-5: HGTs without SSTS and with wrong branch length values. The plots show the posterior probabilities ($P(S_i = k)$, $k = 1, \ldots, 11$) for the eleven tree topologies $T_1$-$T_{11}$.

### 7.3.3 Application

We apply the SSTS algorithm to the ribosomal protein gene *rps11* data described in Chapter 6. Given the underlying species which can be obtained by removing the two HGTs from the network depicted in Figure 6-1, for simplicity we consider a subset of all possible reticulation events, and hence a subset of all possible tree topologies. Specifically twelve topologies including those depicted in Figure 6-2, are considered. The branch lengths are fixed to some values randomly chosen from a $U[0,1]$ with appropriate summation restrictions. The results of the algorithm are reported in Table 7.3 and show that the $M_{1,2}$ is the model visited the most number of times. $M_{1,2}$ is the model which contains the two trees, $T_1$ and $T_2$, depicted in Figures 6-2a and 6-2b. The selection of this model is in line with the results obtained in Chapter 6.

The results of the posterior probabilities for the two topologies, when using the SSTS algorithm combined with the forward-backward procedure, not reported here, are also computed. They are quite close to those reported in Figure 6-4.

## 7.4 Discussion

A more flexible and general algorithm called stochastic search tree selection has been presented in order to avoid exploring the entire space of tree topologies when the reticulations are many and the tree topologies not easily enumerated. This procedure allows us to to restrict the set of reticulations (or of tree topologies) since many of them would contribute little or nothing to the likelihood of the data. The performance of the algorithm has been tested on simulated data showing that this algorithm is able to recover the true model, and has been applied to the ribosomal protein gene *rps11* data, confirming the results obtained in the previous chapter.

However, although this procedure is an extension of the previous one, the flexibility of the algorithm could be improved. One possible idea to develop a more flexible and realistic algorithm would be to allow us to estimate the branch lengths, rather than fixing their values *a priori*. Another possibility would be to make step (c), in the updating of the topology HMM, more flexible. This means that, instead of altering a randomly chosen topology by swapping a tree topology chosen from the trees that are currently in the model for a randomly chosen topology not included in the current model, we could make a local rearrangement of an existing topology by selecting a branch in the tree

| Model | Frequency | Model | Frequency |
|---|---|---|---|
| $M_{1,2}$ | **0.430** | $M_{1,3,4,9}$ | 0.012 |
| $M_{1,3}$ | 0.100 | $M_{1,3,4,10}$ | 0.014 |
| $M_{1,5}$ | 0.002 | $M_{1,3,4,11}$ | 0.001 |
| $M_{1,7}$ | 0.001 | $M_{1,3,5,11}$ | 0.003 |
| $M_{1,12}$ | 0.001 | $M_{1,3,7,9}$ | 0.004 |
| $M_{1,2,3}$ | 0.010 | $M_{1,3,7,10}$ | 0.006 |
| $M_{1,2,5}$ | 0.001 | $M_{1,3,5,11}$ | 0.001 |
| $M_{1,2,4}$ | 0.001 | $M_{1,3,9,10}$ | 0.001 |
| $M_{1,2,5}$ | 0.025 | $M_{1,3,9,11}$ | 0.002 |
| $M_{1,2,12}$ | 0.001 | $M_{1,3,9,12}$ | 0.001 |
| $M_{1,3,6}$ | 0.003 | $M_{1,5,7,9}$ | 0.004 |
| $M_{1,3,7}$ | 0.009 | $M_{1,5,7,10}$ | 0.003 |
| $M_{1,3,9}$ | 0.026 | $M_{1,5,7,12}$ | 0.002 |
| $M_{1,3,10}$ | 0.032 | $M_{1,5,9,12}$ | 0.001 |
| $M_{1,3,12}$ | 0.002 | $M_{1,5,10,12}$ | 0.001 |
| $M_{1,4,5}$ | 0.001 | $M_{1,9,10,12}$ | 0.060 |
| $M_{1,5,6}$ | 0.001 | $M_{1,2,3,7,9}$ | 0.001 |
| $M_{1,5,7}$ | 0.050 | $M_{1,2,4,5,7}$ | 0.002 |
| $M_{1,5,10}$ | 0.006 | $M_{1,2,4,5,9}$ | 0.001 |
| $M_{1,5,12}$ | 0.007 | $M_{1,2,4,9,10}$ | 0.001 |
| $M_{1,7,9}$ | 0.001 | $M_{1,2,5,7,9}$ | 0.004 |
| $M_{1,2,3,5}$ | 0.046 | $M_{1,2,5,7,10}$ | 0.080 |
| $M_{1,2,3,7}$ | 0.002 | $M_{1,3,5,8,10}$ | 0.001 |
| $M_{1,2,3,9}$ | 0.002 | $M_{1,3,5,9,10}$ | 0.001 |
| $M_{1,2,4,5}$ | 0.002 | $M_{1,3,5,9,11}$ | 0.001 |
| $M_{1,2,4,10}$ | 0.001 | $M_{1,3,5,10,12}$ | 0.001 |
| $M_{1,2,5,6}$ | 0.002 | $M_{1,5,7,10,12}$ | 0.001 |
| $M_{1,2,5,7}$ | 0.022 | $M_{1,2,3,5,7,9}$ | 0.001 |
| $M_{1,3,4,8}$ | 0.001 | $M_{1,2,3,5,7,12}$ | 0.001 |

Table 7.3: The frequencies indicate the proportion of times the SSTS algorithm with wrong branch length values has visited the models in the ribosomal protein gene *rps11* data.

and cutting it in a similar manner to that described by Webb *et al*. (2009) and also implicit in the work by Song and Hein (2005). However some caution is required when performing this step. In fact in our setting, not all trees would be consistent with reticulations, and we should also account for the appropriate summation restrictions on the branch lengths.

# Chapter 8

# Conclusions and discussion

## 8.1 Summary

One of the aims of the thesis was to conduct a review of the current state of the art in phylogenetic networks. We have presented a survey of three reticulate network methods at the species level with the aim of providing an accessible introduction to this fascinating field of research, trying to achieve a good balance between mathematical tractability and intuition. In particular the ML estimation, ML combined with HMMs, and the MP approach for phylogenetic networks reconstruction and inference have been reviewed. All these methods are based on the definition of a phylogenetic network as a DAG obtained by positing a set of edges between pairs of the branches of the species tree to model reticulation events. This survey achieved two results: firstly, it offered an accessible introduction to phylogenetic networks at the species level as all these methods have been published in advanced bioinformatics/computational biology journals that cannot be easily accessed, adapted, or applied given the complexity of the topic. Secondly, this review, as opposed to others available in the literature (e.g. Posada and Crandall, 2001; Morrison, 2005; Huson and Bryant, 2006; Makarenkov et al., 2006, to name a few), focuses on phylogenetic network methods at the species level which are based on the simple idea that a network can be naturally and intuitively decomposed into phylogenetic trees. This review does not provide a full overview of the existing literature, rather it provides an overview of more recent methods which have not received much attention in this literature.

The main object of the thesis was the development of a Bayesian modelling framework for phylogenetic networks. To achieve this goal MCMC techniques have been employed. In particular to compute the posterior quantities of the

branch lengths and evolution model parameters, Metropolis-Hastings algorithms have been proposed, whereas to allow inferences to be made regarding the number of different phylogenies for different parts of DNA sequences, two approaches have been considered: naive, where the sites are modelled independently and a first order HMM structure, where the sites are modelled dependently. Also, the stochastic forward-backward algorithm, which is a single component block procedure has been contrasted to the Gibbs sampler, which is a large component block procedure. The performance of the approach has been validated on synthetic data and applied to real data. The simulation results highlighted the importance to model sites dependently by using a first order HMM structure as sites modelled independently might cause the inference of an erroneous phylogenies at sites. Also, the results showed that the Gibbs sampling performance is poor as compared to the forward-backward algorithm in terms of convergence and mixing. To investigate the impact of some model misspecifications and provide some practical insights, several simulation scenarios have been presented. For example depending on the problem at hand the choice of the model of evolution might not be so crucial as acceptable tree classifications can still be obtained by using simpler models. Also the MCMC algorithm might not work well when trees are confounded as they distort the results in the same way as highly collinear covariates can cause regression parameters to be inefficient or make them quite unstable. As for the application to the *rps11* data the findings showed that significant variation caused by reticulation events is detected. Finally, a more general and flexible algorithm which avoids exploring the entire space of tree topologies when the reticulations are many has been proposed. The SSTS allows the data to dictate how many phylogenies are required to explain the data. This has been achieved by adapting the stochastic search variable selection algorithm to the phylogenetic network framework. The performance of the algorithm has been tested on simulated data showing that this algorithm is able to recover the true model, and has been applied to the ribosomal protein gene *rps11* data, confirming the previous results. When using the proposed algorithm, the selection of a set of trees is of extreme importance; we saw that we can rely on previous findings and biological reasoning. However sometimes a more automatic and less ad hoc method could also be used; its advantage would be more evident when $2^K$ is a large number, and a priori one tree is not preferred over others.

## 8.2 Extensions and future work

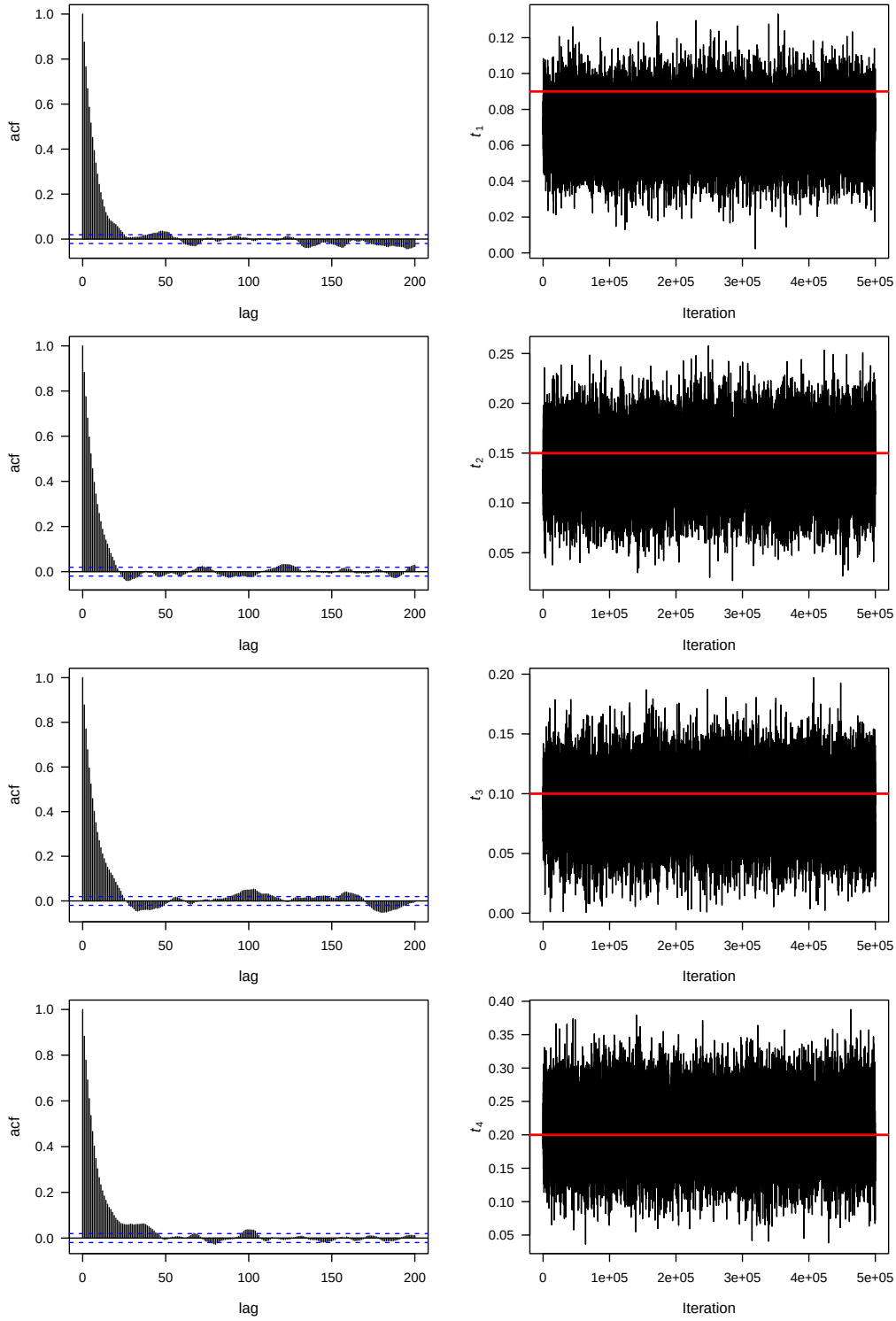
Although the SSTS procedure is a more general algorithm, its flexibility could be further improved. As already discussed in Chapter 7 a more realistic version could be developed to allow us to estimate the branch lengths, and to update the topology HMM, using a local rearrangement of an existing topology in the same way as that described by Webb *et al*. (2009). However some caution is required when trying to generalise the algorithm as the appropriate summation restrictions have to be imposed on the branch lengths.

Alternatively, the algorithm could be made more general and more realistic by inferring the reticulation events, given an underlying species tree. This could be achieved by using a birth and death MCMC (BDMCMC) formulation. BDMCMC has been developed by Stephens (2000) in the context of mixture models with unknown number of components. This technique simulates a birth-death process in which parameters (components) are added (birth) and deleted (deaths). This process is run for a fixed length of time within each iteration in order to update the model. Thus, many models may be visited during the continuous time process run within each iteration. An important feature of the BDMCMC sampling is that a continuous time jump process is associated with the birth-and-death rates: whenever a jump occurs, the corresponding move is always accepted. The acceptance probability of usual MCMC methods is replaced by the differential holding times. In particular, implausible configurations die quickly. By following this approach, we could turn the problem of inferring mixture models with unknown number of components into a problem of inferring phylogenetic trees with unknown number of reticulations by making the necessary modifications. Alternative algorithms to BDMCMC, such as the reversible jump MCMC (RJMCMC) algorithm, could also be used. However, in contrast to RJMCMC, the BDMCMC moves take the advantage of the natural nested structure of the models, removing the need for the calculation of a complicated Jacobian, and making the implementation more straightforward. Also, the BDMCMC moves do not make use of any parameter constraint and of any missing data structure.

In conclusion, although several methodological improvements are necessary to provide more general and reliable results, and this will be our direction for future research, we should recognise that the modelling and estimation of phylogenetic networks is a relatively young field of research; thus we still lack a comprehensive theoretical framework and empirical investigation. Nonetheless we believe that the approach discussed in this thesis is a step forward in

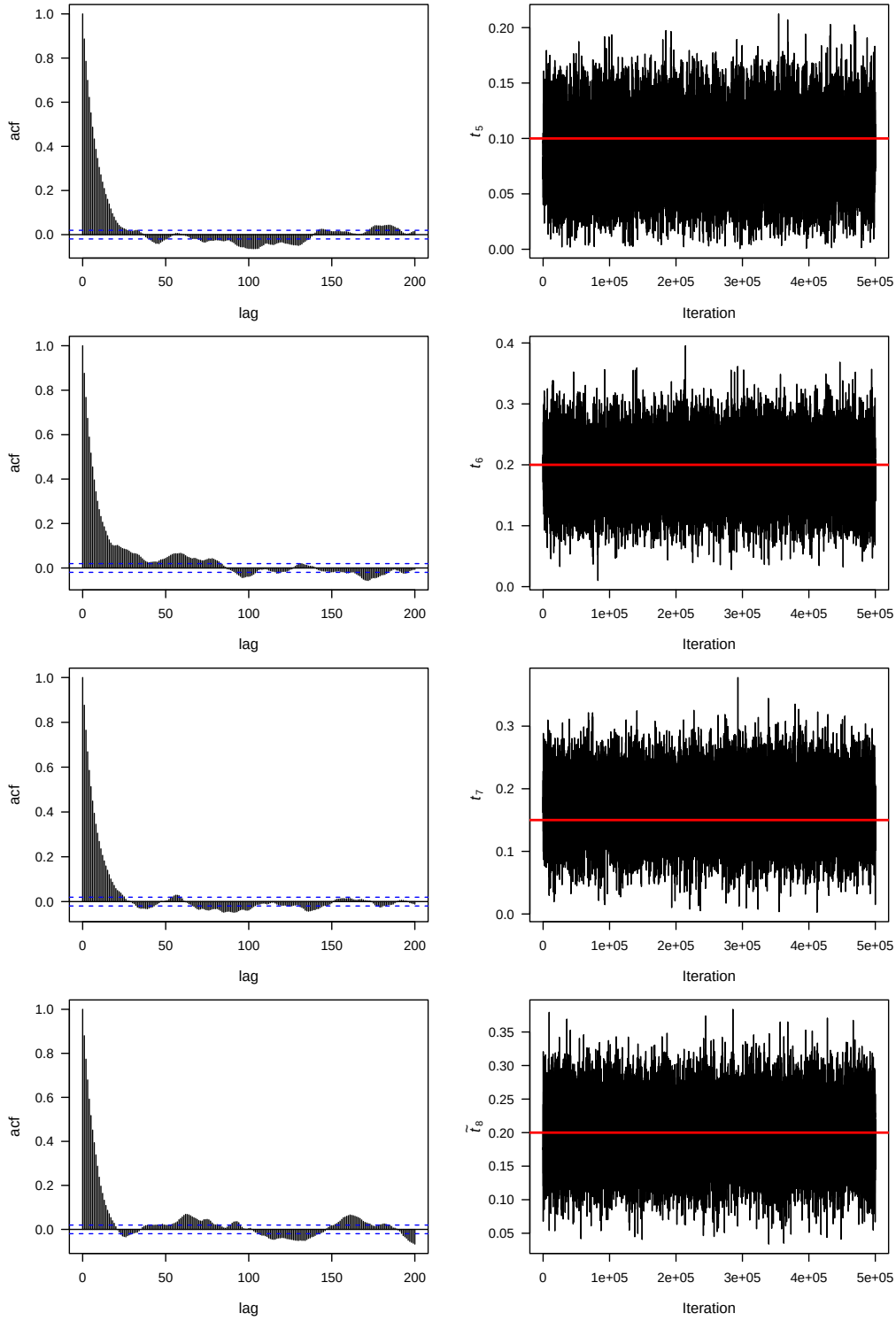
this direction, and our developments demonstrate the power and flexibility of phylogenetic network modelling as a means of accounting for non-treelike events, identifying the phylogenies underlying biomolecular data.

# Appendix



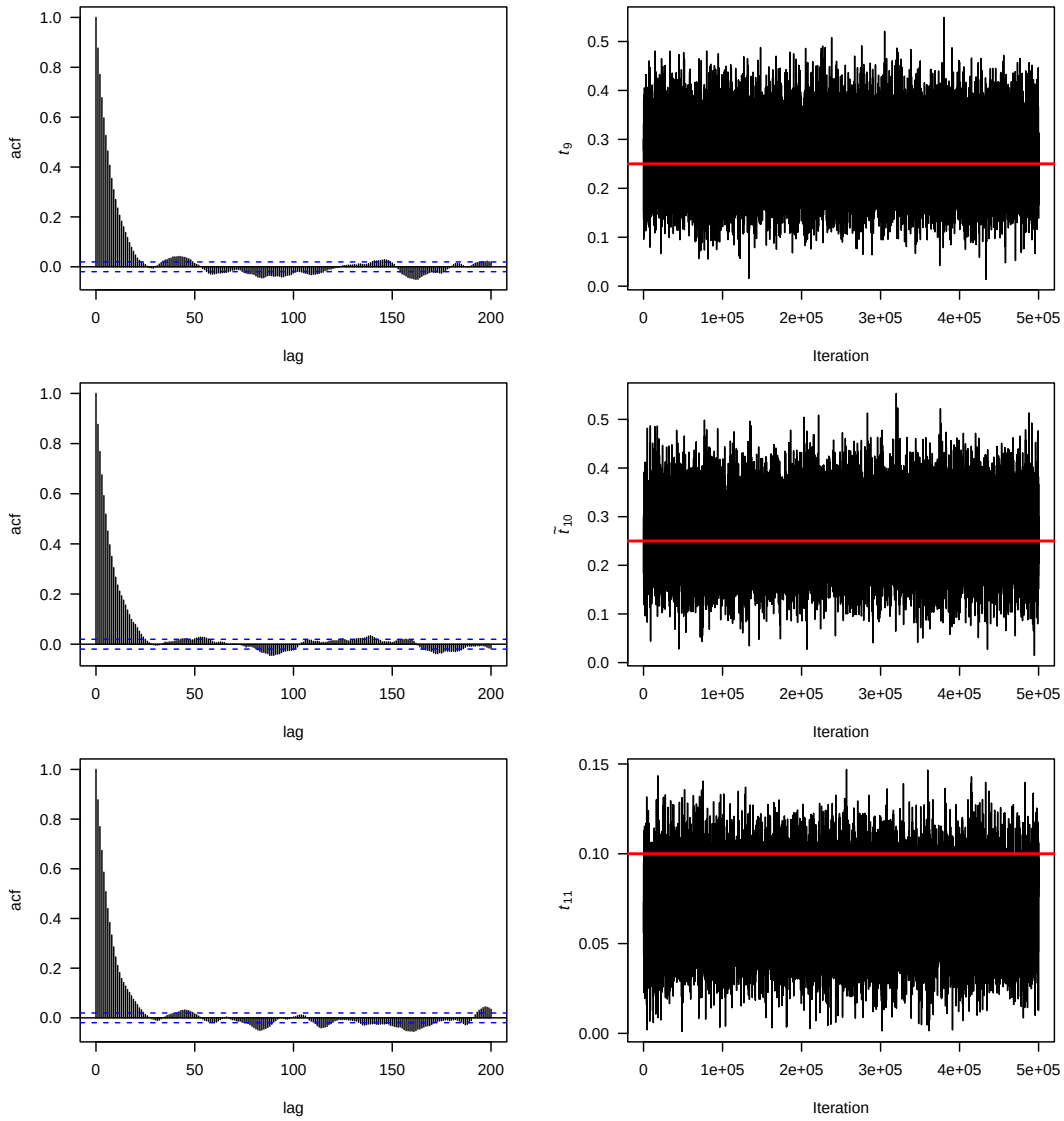

(a) Autocorrelation function and trace plots for the branch edges $t_1$-$t_4$ compared to their true values (red line).

Figure 8-1: Autocorrelation function and trace plots for the branch edges with HMM on **S**.



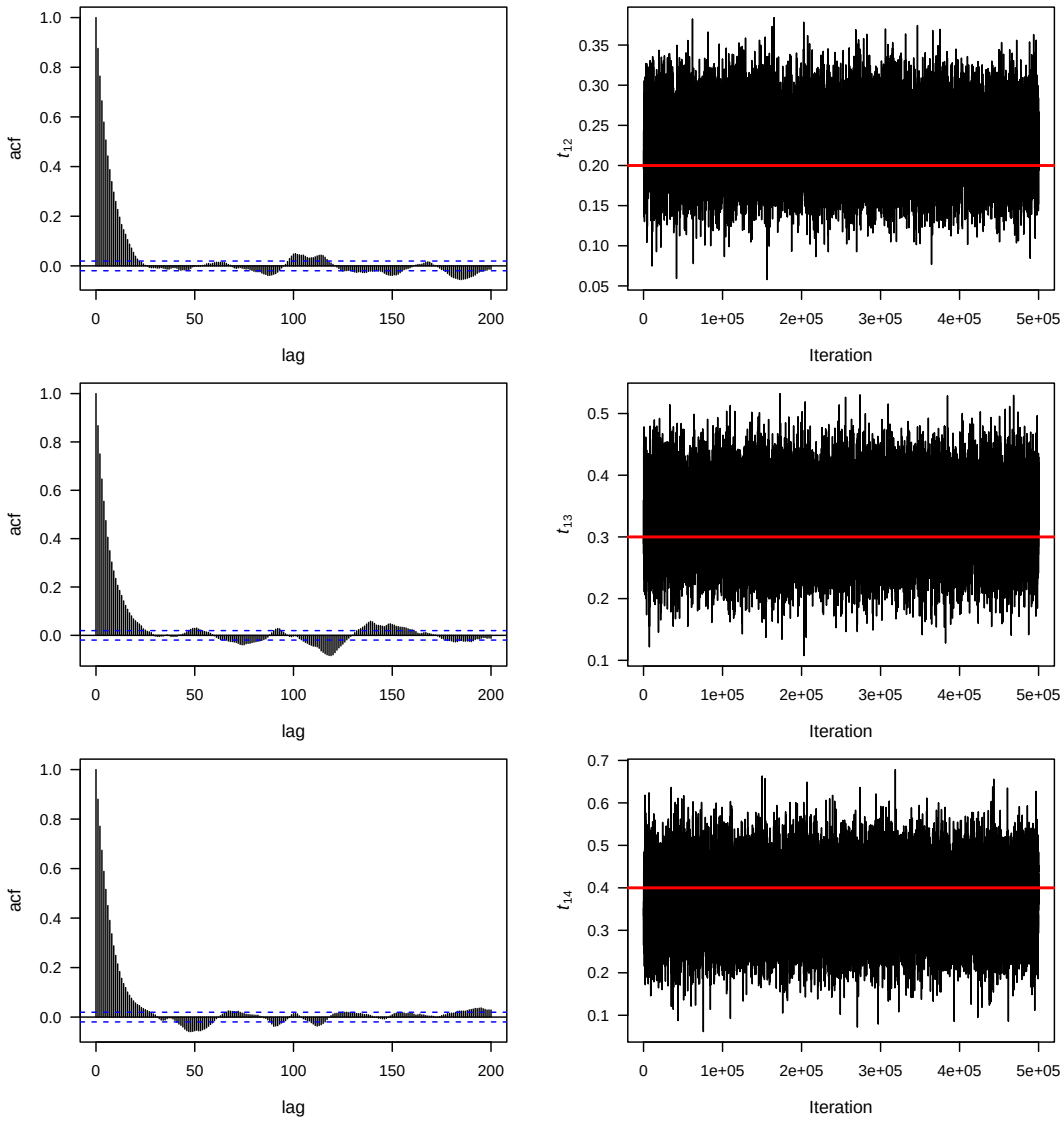

(b) Autocorrelation function and trace plots for the branch edges $t_5$-$t_7$ and $\tilde{t}_8$ compared to their true values (red line).

Figure 8-1: Autocorrelation function and trace plots for the branch edges with HMM on **S** (con't).




(c) Autocorrelation function and trace plots for the branch edges $t_9, \tilde{t}_{10}, t_{11}$ compared to their true values (red line).

Figure 8-1: Autocorrelation function and trace plots for the branch edges with HMM on **S** (con't).




(d) Autocorrelation function and trace plots for the branch edges $t_{12}$-$t_{14}$ and compared to their true values (red line).

Figure 8-1: Autocorrelation function and trace plots for the branch edges with HMM on **S** (con't).

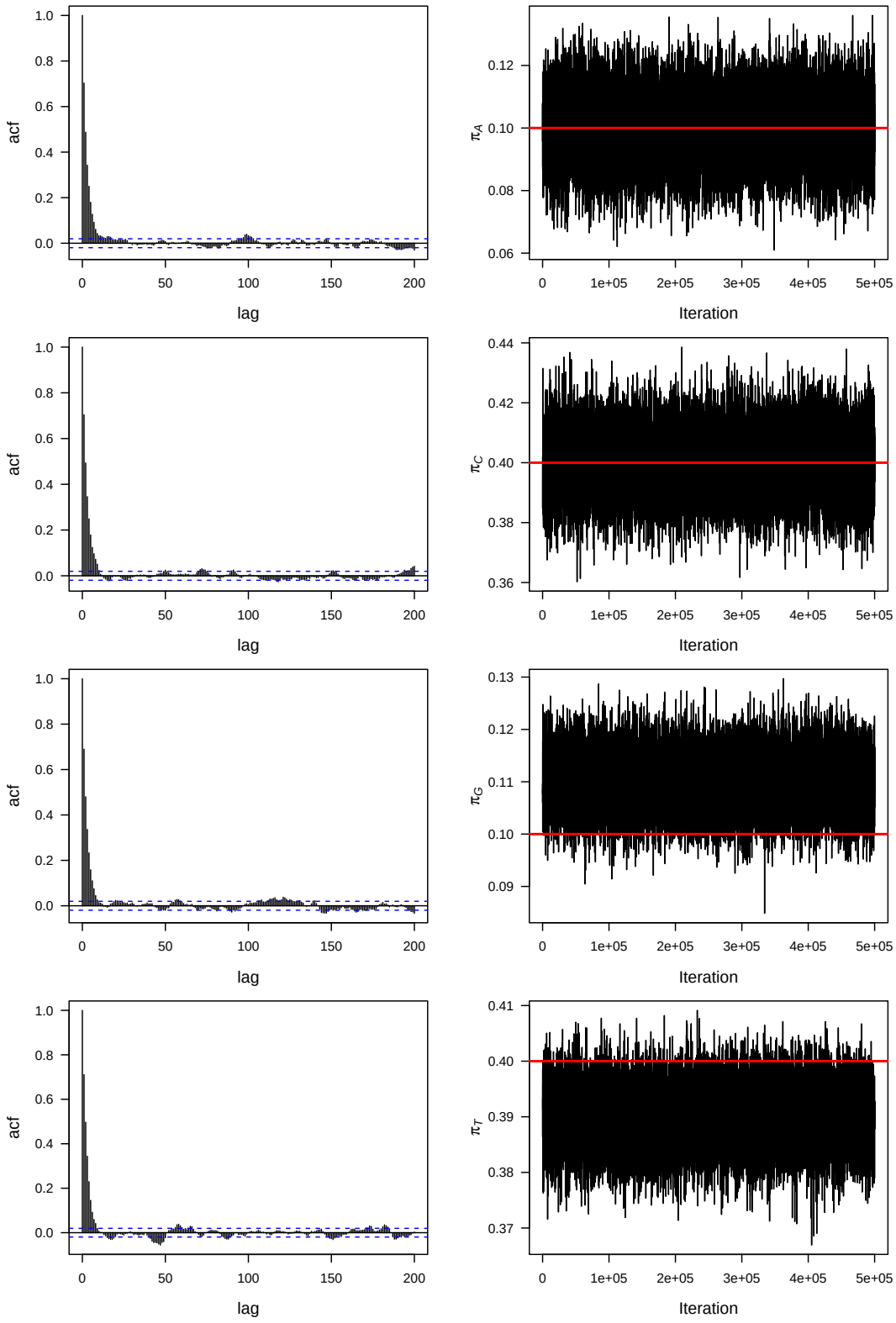
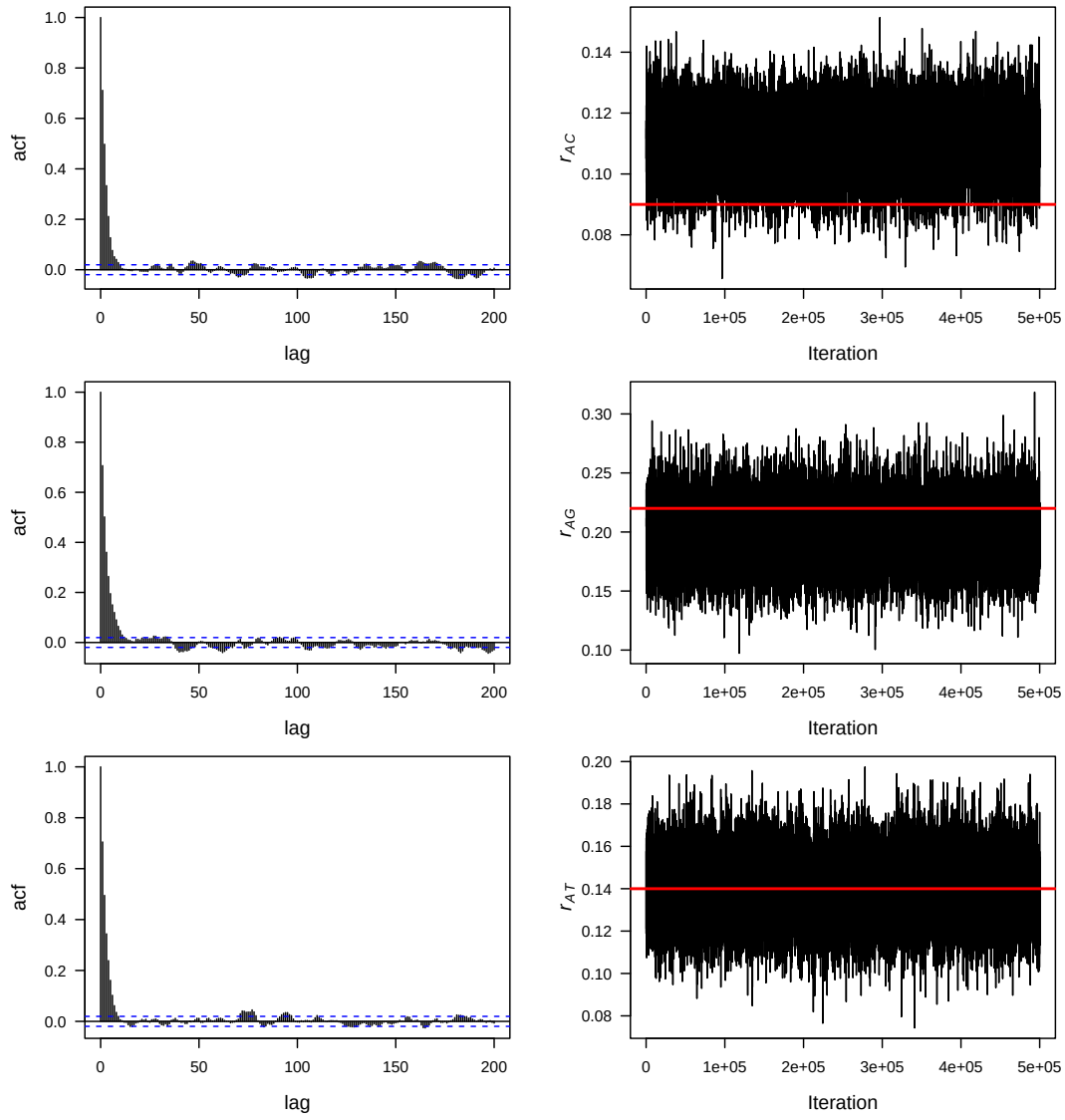
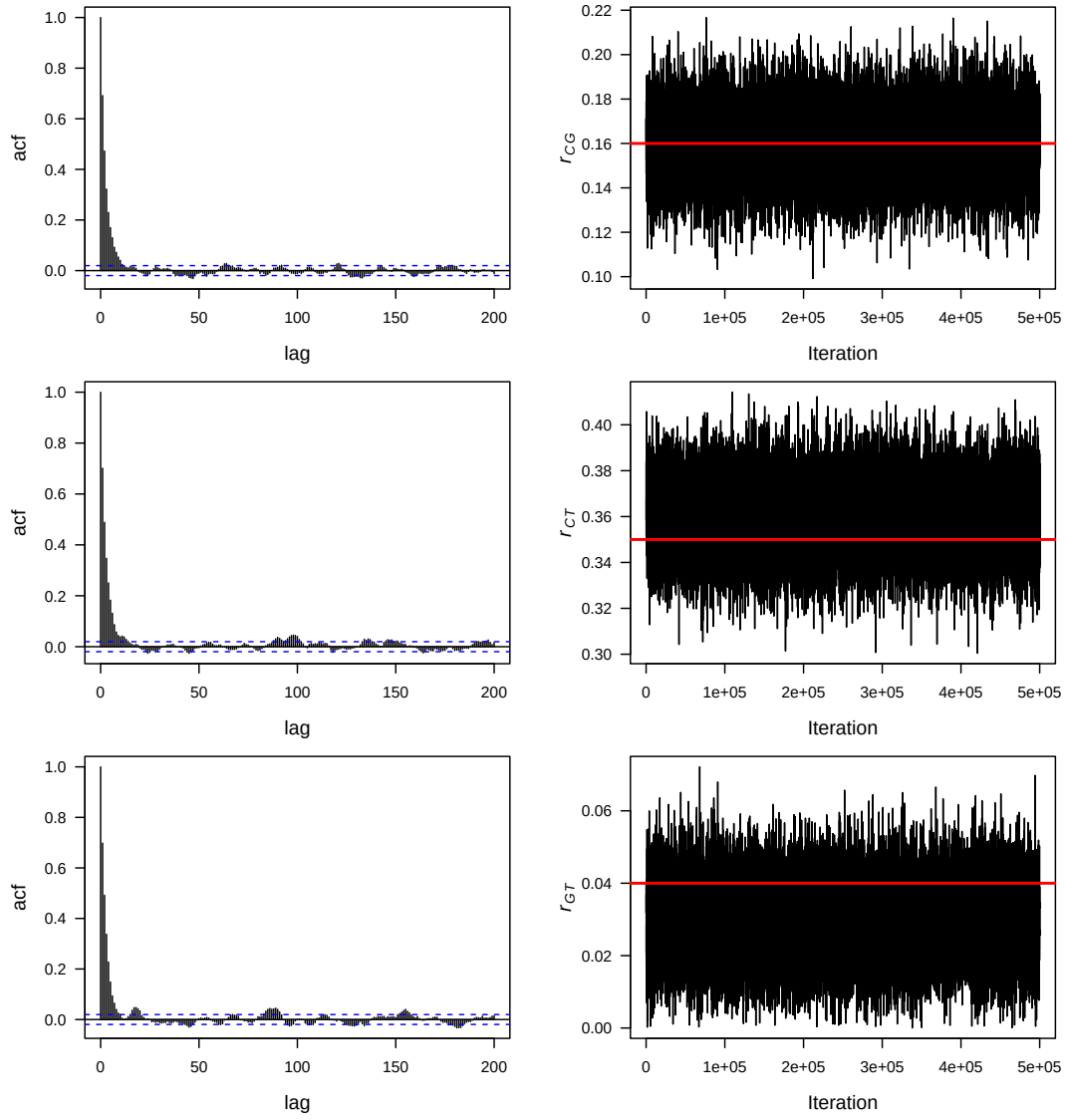



Figure 8-2: Autocorrelation function and trace plots for the nucleotide frequencies $\pi_A, \pi_C, \pi_G, \pi_T$ compared to their true values (red line).




(a) Autocorrelation function and trace plots for the rates of substitution $r_{AC}, r_{AG}, r_{AT}$ compared to their true values (red line).

Figure 8-3: Autocorrelation function and trace plots for the rates of substitution.




(b) Autocorrelation function and trace plots for the rates of substitution $r_{CG}, r_{CT}, r_{GT}$ compared to their true values (red line).

Figure 8-3: Autocorrelation function and trace plots for the rates of substitution (con't).

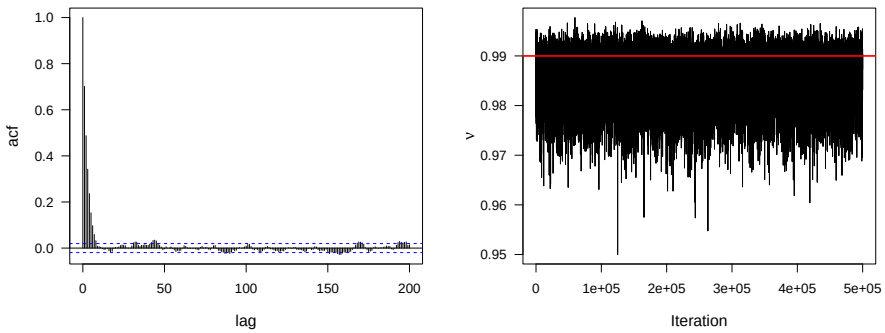
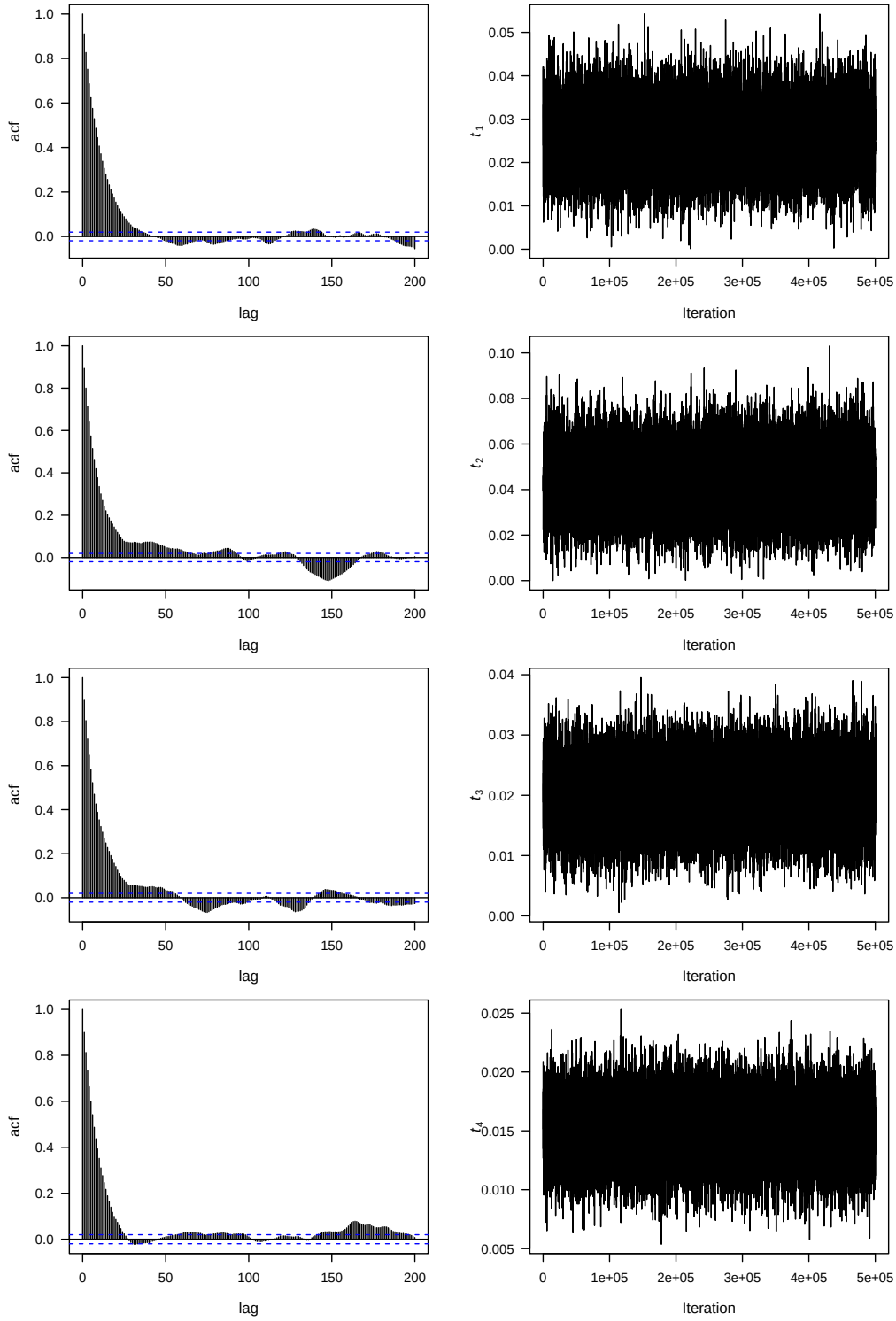
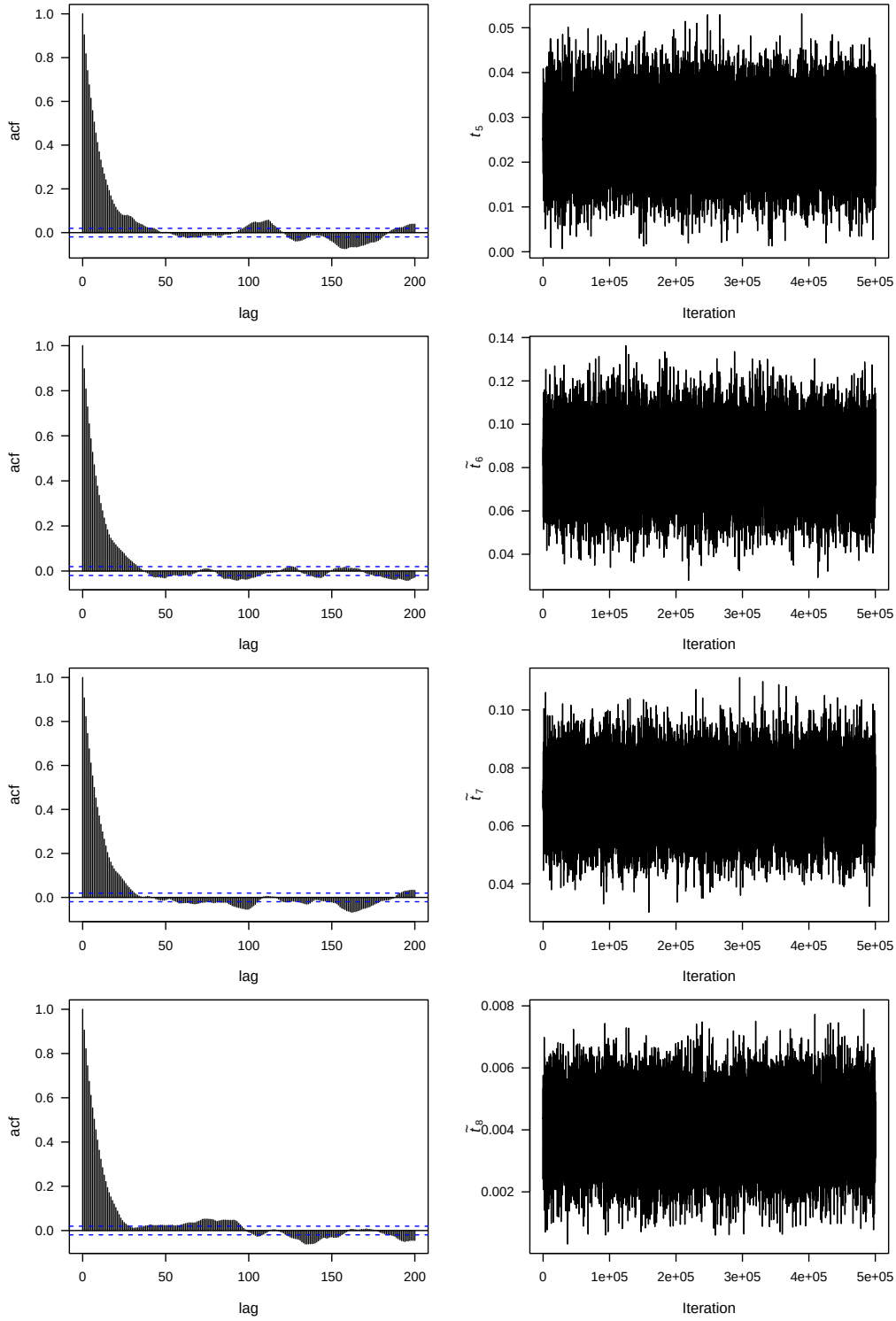



Figure 8-4: Autocorrelation function and trace plots for the probability of not changing topology $\nu$.



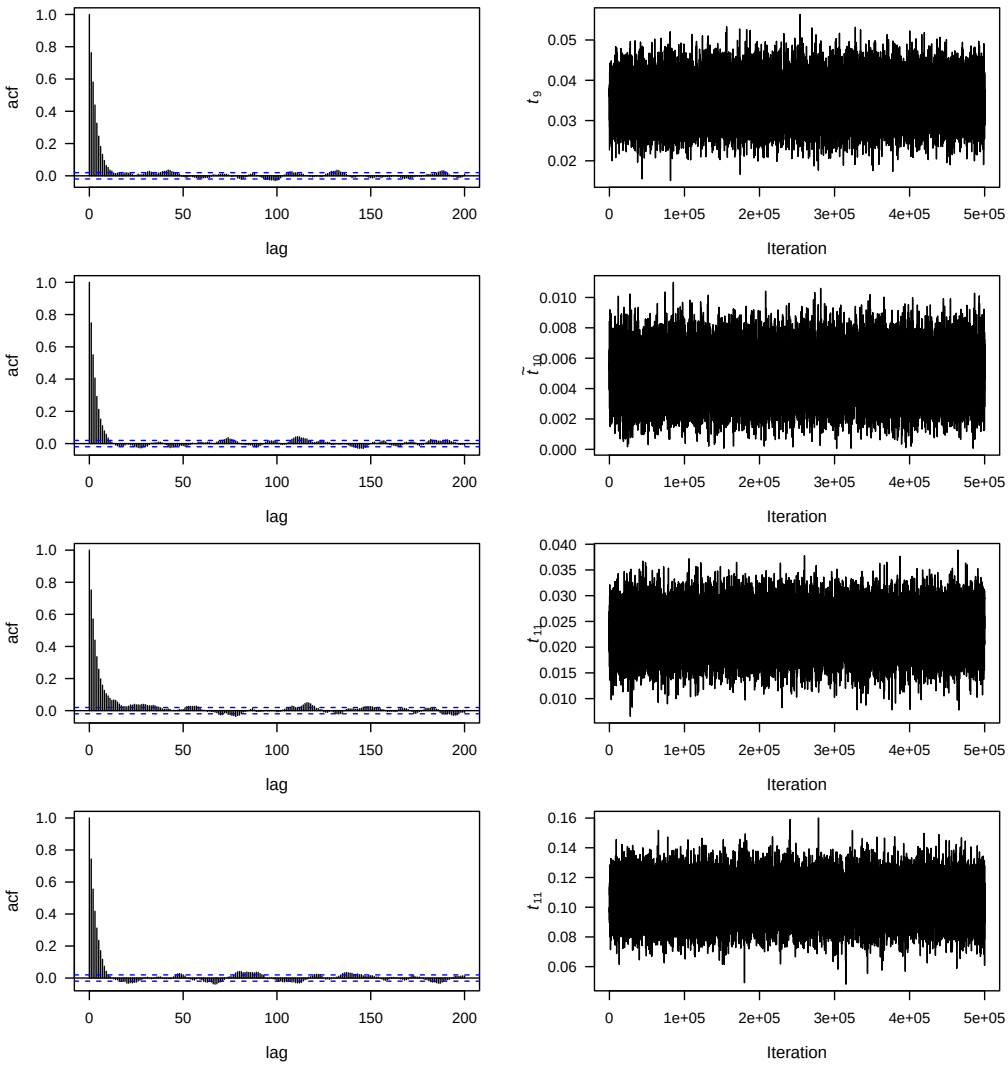

(a) Autocorrelation function and trace plots for the branch edges $t_1$-$t_4$.

Figure 8-5: Autocorrelation function and trace plots for the branch edges with HMM on **S**.




(b) Autocorrelation function and trace plots for the branch edges $t_5$, $\tilde{t}_6$, $\tilde{t}_7$ and $t_8$.

Figure 8-5: Autocorrelation function and trace plots for the branch edges with HMM on **S** (con't).




(c) Autocorrelation function and trace plots for the branch edges $t_9$, $\tilde{t}_{10}$, $t_{11}$ and $t_{12}$.

Figure 8-5: Autocorrelation function and trace plots for the branch edges with HMM on **S** (con't).

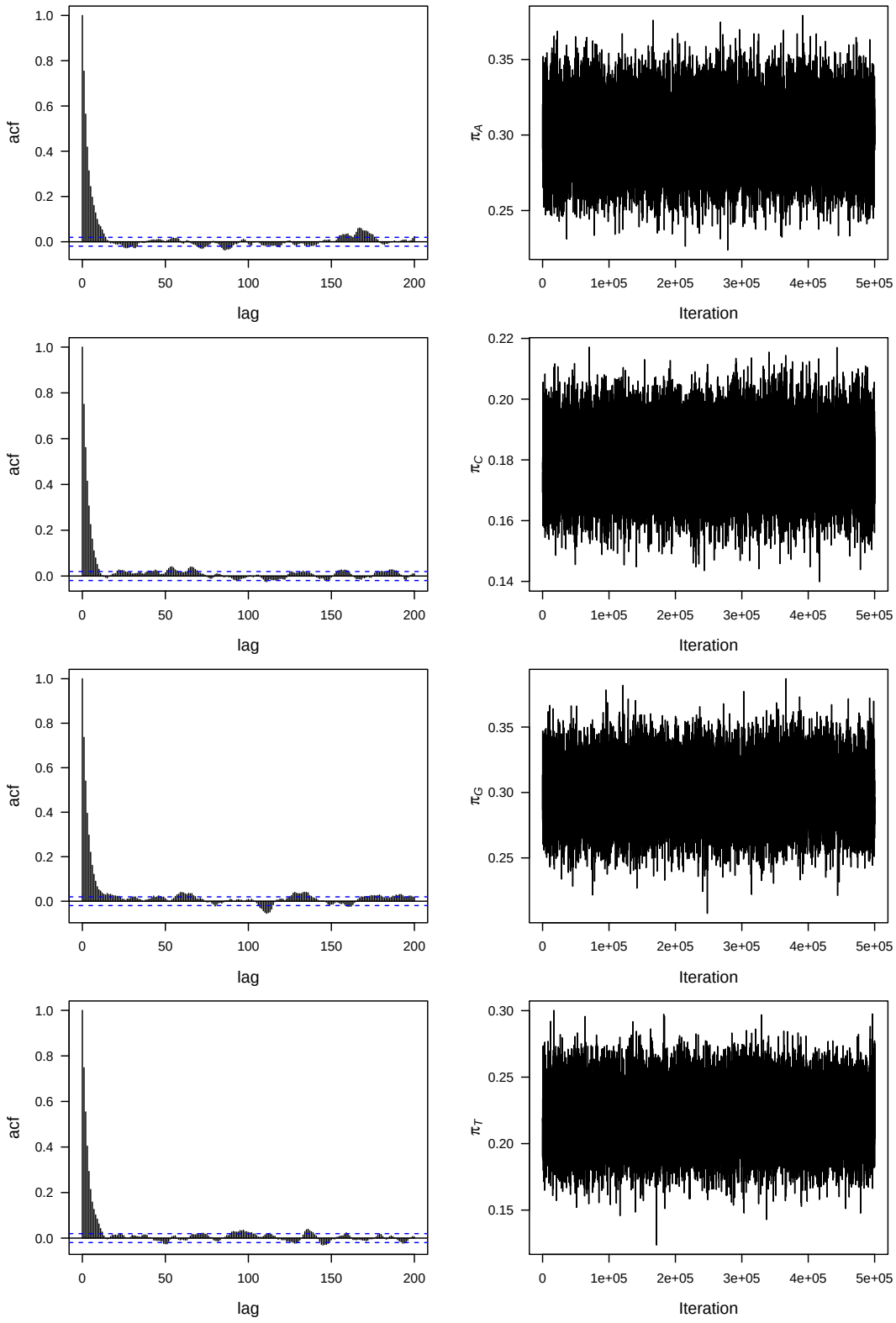
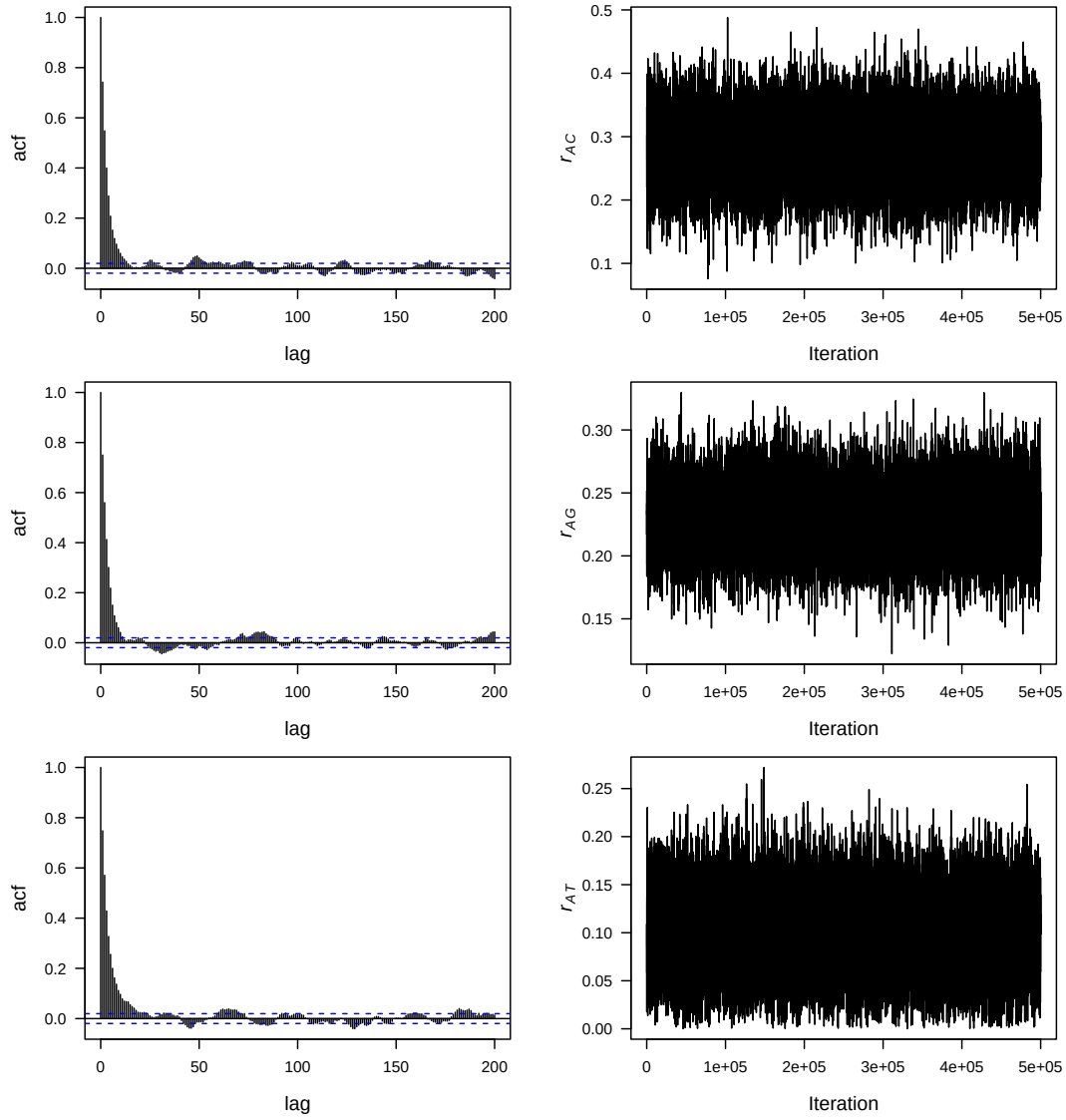
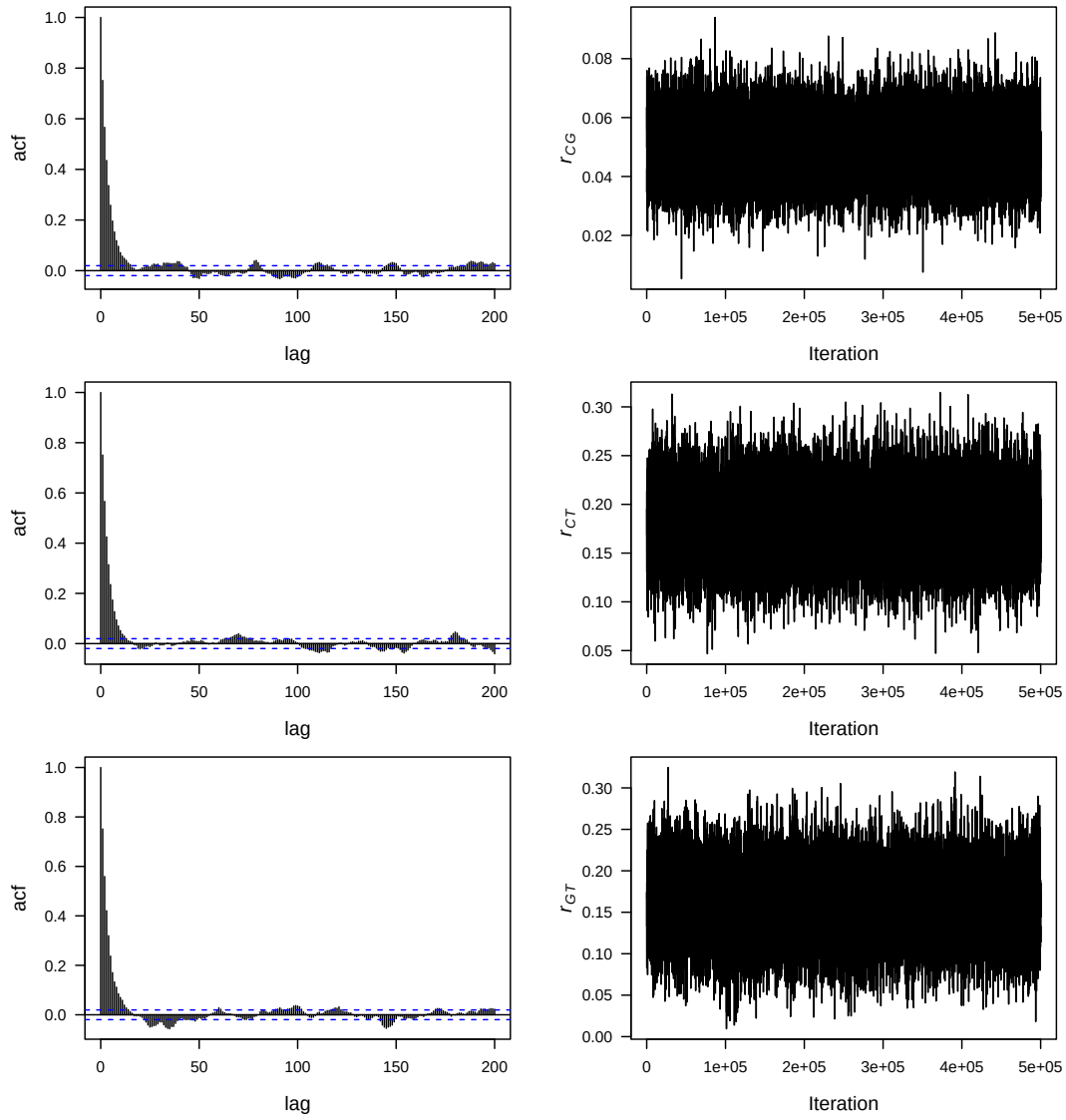



Figure 8-6: Autocorrelation function and trace plots for the nucleotide frequencies $\pi_A, \pi_C, \pi_G, \pi_T$.




(a) Autocorrelation function and trace plots for the rates of substitution $r_{AC}, r_{AG}, r_{AT}$.

Figure 8-7: Autocorrelation function and trace plots for the rates of substitution.




(b) Autocorrelation function and trace plots for the rates of substitution $r_{CG}, r_{CT}, r_{GT}$.

Figure 8-7: Autocorrelation function and trace plots for the rates of substitution (con't).

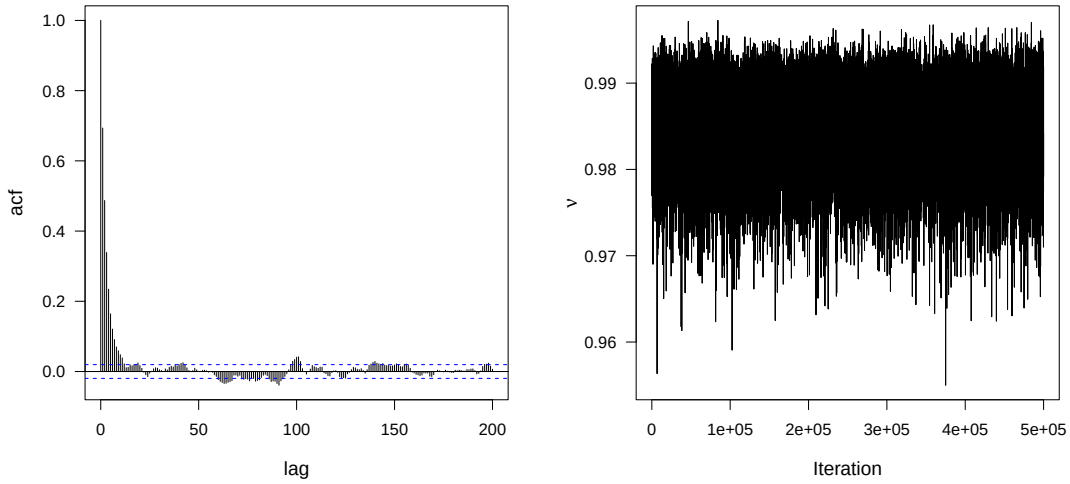



Figure 8-8: Autocorrelation function and trace plots for $\nu$.

# Bibliography


[1] BANDELT, H. J., DRESS, A. W. M., 1992. A canonical decomposition theory for metrics on a finite set. *Advances in Mathematics*, 92(1), pp.47–105.

[2] BANDELT, H. J., FORSTER, P., SYKES, B. C., RICHARDS, M. B., 1995. Mitochondrial portraits of human population using median networks. *Genetics*, 141(2), pp.743–753.

[3] BANDELT, H. J., MACAULAY, V., RICHARDS, M. B, 2000. Median networks: speedy construction and greedy reduction, one simulation, and two case studies from human mtDNA. *Molecular Phylogenetics and Evolution*, 16(1), pp.8–28.

[4] BERGTHORSSON, U., ADAMS, K. L., THOMASON, B., PALMER, J. D., 2003. Widespread horizontal transfer of mitochondrial genes in flowering plants. *Nature*, 424, pp.197–201.

[5] BERGTHORSSON, U., RICHARDSON, A. O., YOUNG, G. J., GOERTZEN, L. R., PALMER, J. D., 2004. Massive horizontal transfer of mitochondrial genes from diverse land plant donors to the basal angiosperm Amborella. *Proceedings of the National Academy of Sciences of the United States of America*, 101(51), pp.17747-17752.

[6] BESAG, J., 1974. Spatial interaction and the statistical analysis of lattice systems (with discussions). *Journal of the Royal Statistical Society Series B*, 36(2), pp.192–236.

[7] BOYS, R. J., HENDERSON, D. A., WILKINSON, D. J., 2000. Detecting homogeneous segments in DNA sequences by using hidden Markov models. *Journal of the Royal Statistical Society C*, 49(2), pp.269–285.

[8] BROOKS, P., GELMAN, A., 1998. General methods for monitoring convergence of iterative simulations. *Journal of Computational and Graphical Statistics*, 7(4), pp.434–455.

[9] BRYANT, D. MOULTON, V., 2004. Neighbor-net: An agglomerative method for the construction of planar phylogenetic networks. *Molecular Biology and Evolution*, 21(2), pp.255-265.

[10] CASELLA, G., GEORGE, E. I., 1992. Explaining the Gibbs sampler. *The American Statistician*, 46(3), pp.167-174.

[11] CHIB, S., GREENBERG, E., 1995. Understanding the Metropolis-Hastings algorithm. *The American Statistician*, 49(4), pp.327-335.

[12] COWLS, M. K., CARLIN, B. P., 1996. Markov Chain Monte Carlo convergence diagnostics: A comparative review. *Journal of the American Statistical Association*, 91(434), pp.883–904.

[13] DENISON, D. G. T., MALLICK, B. K., SMITH, A. F. M., 1998. Automatic Bayesian curve fitting. *Journal of the Royal Statistical Society B*, 60(2), pp.333–350.

[14] FELSENSTEIN, J., 1981. Evolutionary trees from DNA sequences: a maximum likelihood approach. *Journal of Molecular Evolution*, 17(6), pp.368–376.

[15] FELSENSTEIN, J., 2004. Inferring Phylogenies. USA: Sinauer Associates.

[16] FELSENSTEIN, J., 2009. The Newick tree format. Department of Genome Sciences, University of Washington, Seattle, USA.

[17] FITCH, W. M., 1971. Toward defining the course of evolution: minimum change for a specific tree topology. *Systematic Zoology*, 20(4), pp.406-416.

[18] GEORGE, E. I., McCULLOCH, R. E., 1993. Variable selection via Gibbs sampling. *Journal of the American Statistical Association*, 88(423), pp.881–889.

[19] GEORGE, E. I., McCULLOCH, R. E., TSAY, R. S., 1996. Two approaches for Bayesian model selection with applications. *In*: D. A. BERRY, M. CHALONER, J. K. GEWEKE. *Bayesian Analysis in Statistics and Econometrics*. New York, pp.339–348.

[20] GEYER, C. J., 1992. Practical Markov Chain Monte Carlo. *Statistical Science*, 7(4), pp.473-511.

[21] GILKS, W. R., RICHARDSON, S., SPIEGELHALTER, D. J., 1996. Introducing Markov chain Monte Carlo. *In*: W. R. GILKS, S. RICHARDSON, D. J. SPIEGELHALTER. *Markov Chain Monte Carlo in Practice*. London: Chapman and Hall, pp.89–114.

[22] GOLUB, G. H., van LOAN, C. F., 1996. *Matrix computations*. 3th ed. Baltimore: Johns Hopkins University Press.

[23] GREEN, P. J., 1995. Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 82(4), pp.711–732.

[24] GREEN, P. J., HAN, X.-L., 1992. Metropolis methods, Gaussian proposals, and antithetic variables. *Lecture Notes in Statistics*, 74, pp.142-164.

[25] GRIFFITHS, R. C., MARJORAM, P., 1996. Ancestral inference from samples of DNA sequences with recombination. *Journal of Computational Biology*, 3(4), pp.479–502.

[26] HARTIGAN, J. A., 1973. Minimum mutation fits to a given tree. *Biometrics*, 29(1), pp.53-65.

[27] HEIN J., 1990. Reconstructing evolution of sequences subject to recombination using parsimony. *Mathematical Biosciences*, 98(2), pp.185-200.

[28] HEIN J., 1993. A heuristic method to reconstruct the history of sequences subject to recombination. *Journal of Molecular Evolution*, 36(4), pp.396-405.

[29] HEUER H., SMALLA K., 2007. Horizontal gene transfer between bacteria. *Environmental Biosafety Research* , 6(1-2), pp.3-13.

[30] HOLLAND, B., MOULTON, V., 2003. Consensus networks: A method for visualizing incompatibilities in collections of trees. *Proceedings of Workshops on Algorithms in Bioinformatics*, 2003, pp.165–176.

[31] HUDSON, R. R., 1983. Properties of a neutral allele model with intragenic recombination. *Theoretical Population Biology*, 23(2), pp.183–201.

[32] HUELSENBECK, J. P., RONQUIST, F., 2001. MRBAYES: Bayesian inference of phylogeny. *Bioinformatics* 17(8), pp.754–755.

[33] HUSMEIER, D., McGUIRE, G., 2002. Detecting recombination with MCMC. *Bioinformatics*, 18(1), pp.S345–S353.

[34] HUSON, D. H., BRYANT, D., 2006. Application of phylogenetic networks in evolutionary studies. *Molecular Biology and Evolution*, 23(2), pp.254–267.

[35] JIN, G., NAKHLEH, L, SNIR, S., TULLER T., 2006. Maximum likelihood of phylogenetic networks. *Bioinformatics*, 22(21), pp.2604–2611.

[36] JIN, G., NAKHLEH, L, SNIR, S., TULLER T., 2007a. Efficient parsimony-based methods for phylogenetic network reconstruction. *Bioinformatics*, 23(2), pp.e123–e128.

[37] JIN, G., NAKHLEH, L, SNIR, S., TULLER T., 2007b. Inferring phylogenetic networks by the maximum parsimony criterion: a case study. *Molecular Biology and Evolution*, 24(1), pp.324–337.

[38] JIN, G., NAKHLEH, L, SNIR, S., TULLER T., 2009. Parsimony score of phylogenetic networks: Hardness results and a linear-time heuristic. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 6(3), pp.495–505.

[39] JUKES, T. H., CANTOR, C. R., 1969. Evolution of protein molecules. *In*: H. N. MUNRO. *Mammalian Protein Metabolism*. New York: Academic Press, pp. 21–132.

[40] LARGET, B., SIMON, D. L., 1999. Markov chain Monte Carlo algorithms for the Bayesian analysis of phylogenetic trees. *Molecular Biology and Evolution*, 16(6), pp.750–759.

[41] LINDER, C. R., MORET, B. M. E., NAKHLEH, L., WARNOW, T., 2004. Network (reticulate) evoulution: biology, models, and algorithms. *Proceedings of the Pacific Symposium on Biocomputing*, 2004, Hawaii.

[42] LINDER, C. R., RIESEBERG, L. H., 2004. Reconstructing patterns of reticulate evolution in plants. *American Journal of Botany*, 91(10), pp.1700-1708.

[43] LINZ, S., RADTKE, A., von HAESELER, A., 2007. A Likelihood framework to measure horizontal gene transfer. *Molecular Biology and Evolution*, 24(6), pp.1312-1319.

[44] LOZA-REYES, E., 2010. *Classification of phylogenetic data via Bayesian mixture modelling*. PhD thesis, University of Bath, UK.

[45] MAKARENKOV, V., KEVORKOV, D., LEGENDRE, P., 2006. Phylogenetic network reconstruction approaches. *In*: D. K. ARORA, R. M. BERKA,

G. B. SINGH. *Applied Mycology and Biotechnology*. Netherlands: Elsevier, pp.61–97.

[46] MARTÍNEZ, J. L., BAQUERO, F., ANDERSSON, D. I., 2007. Predicting antibiotic resistance. *Nature Reviews in Microbiology*, 5, pp.958–965.

[47] MAU, B., NEWTON, M. A., LARGET, B., 1999. Bayesian phylogenetic inference via Markov chain Monte Carlo methods. *Biometrics*, 55(1),pp.1-12.

[48] MORET, B. M. E., NAKHLEH, L., WARNOW, T., LINDER, C. R., THOLSE, A., PADOLINA, A., SUN, J., TIMME, R., 2004. Phylogenetic networks: modeling, reconstructibility, and accuracy. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 1(1), pp.13–23.

[49] MORRISON, D. A., 2005. Networks in phylogenetic analysis: new tools for population biology. *International Journal for Parasitology*, 35(5), pp.567–582.

[50] NAKHLEH, L., JIN, G., ZHAO, F., MELLOR-CRUMMEY, J., 2005. Reconstructing phylogenetic networks using maximum parsimony. *Proceedings of the IEEE Computational Systems Bioinformatics Conference*, August, 2005, Stanford University, pp.93–102.

[51] POSADA, D., CRANDALL, K. A., 2001. Intraspecific gene genealogies: trees grafting into networks. *Trends in Ecology and Evolution*, 16(1), pp.37–45.

[52] RAMBAUT, A., GRASSLY, N. C., 1997. Seq-Gen: an application for the Monte Carlo simulation of DNA sequence evolution along phylogenetic trees. *Computer Applications in the Biosciences*, 13(3), pp.235-238.

[53] REICH, B. J., STORLIE, C. B., BONDELL, H. D., 2009. Variable selection in Bayesian smoothing spline ANOVA models: Application to deterministic computer codes. *Technometrics*, 51(2), pp.110-120.

[54] RICHARDSON, A. O., PALMER, J. D., 2007. Horizontal gene transfer in plants, *Journal of Experimental Botany*, 58(1), pp.1-9.

[55] RONQUIST, F., HUELSENBECK, J. P, van der MARK, P., 2005. MrBayes v. 3.1 Manual.

[56] RUE, H., HELD, L., 2005. *Gaussian Markov random fields: theory and applications*. London: Chapman & Hall/CRC.

[57] SEMPLE, C., STEEL, M., 2003. *Phylogenetics*. New York: Oxford University Press.

[58] SIEPEL, A., HAUSSLER, D., 2005. Phylogenetic hidden Markov models. *In*: R. NIELSEN. *Statistical Methods in Molecular Evolution*. New York: Springer, pp. 325-351.

[59] SNIR, S., TULLER, T., 2009. The NET-HMM approach: Phylogenetic network inference by combining maximum likelihood and hidden Markov models. *Journal of Bioinformatics and Computational Biology*, 7(4), pp.625–644.

[60] SONG, Y., S., HEIN, J., 2005. Constructing minimal ancestral recombination graphs. *Journal of Computational Biology*, 12(2), pp.147–169.

[61] STEPHENS, M., 2000. Bayesian analysis of mixture models with an unknown number of components: An alternative to reversible jump methods. *The Annals of Statistics*, 28(1), pp.40-74.

[62] STRIMMER, K., WIUF, C., MOULTON, V., 2001. Recombination analysis using directed graphical models. *Molecular Biology and Evolution*, 18(1), pp.97–99.

[63] SUCHARD, M. A., 2005. Stochastic models for horizontal gene transfer: Taking a random walk through tree space. *Genetics*, 170(1),pp.419-431.

[64] TAVARÉ, S., 1986. Some probabilistic and statistical problems in the analysis of DNA sequences. *In*: R. M. MIURA. *Lectures on Mathematics in the Life Sciences*. New York: American Mathematical Society, pp.57–86.

[65] VIALLEFONT, V., RICHARDSON, S.,GREEN, P. J., 2002. Bayesian analysis of poisson mixtures. *Workshop on Statistical Models and Methods for Discontinuous Phenomena*, 14, pp.181–202.

[66] VERBYLA, K. L., HAYES, B. J., BOWMAN, P. J., GODDARD, M. E., 2009. Accuracy of genomic selection using stochastic search variable selection in Australian Holstein Friesian dairy cattle. *Genetical Research*, 91(5), pp.307–311.

[67] WEBB, A., HANCOCK, J. M., HOLMES, C. C., 2009. Phylogenetic inference under recombination using bayesian stochastic topology selection. *Bioinformatics*, 25(2), pp.197-203.

[68] WILLSON, S. J., 2010. Regular networks can be uniquely constructed from their trees. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, to appear.

[69] WON, H., RENNER S. S., 2003. Horizontal gene transfer from flowering plants to Gnetum. *Proceedings of the National Academy of Sciences of the United States of America*, 100(19), pp.10824-10829.

[70] YANG, Z., RANNALA, B., 1997. Bayesian phylogenetic inference using DNA sequences: a Markov chain Monte Carlo method. *Molecular Biology and Evolution*, 14(7), pp.717-724.

[71] YANG, Z., 2006. *Computational Molecular Evolution*. New York: Oxford University Press.

[72] YI, N., GEORGE, V., ALLISON, D. B., 2003. Stochastic search variable selection for identifying multiple quantitative trait loci. *Genetics*, 164(3), pp.1129-1138.

# Further Reading

BROOKS, S. P., 1998. Markov chain Monte Carlo method and its application. *Journal of the Royal Statistical Society D*, 47(1), pp.69-100.

CAVALLI-SFORZA, L. L., EDWARDS, A. W. F., 1967. Phylogenetic analysis: models and estimation procedures. *Evolution*, 21, pp.550-570.

FITCH, W. M., 1997. Networks and viral evolution. *Journal of Molecular Evolution*, 44(1), pp.S65–S75.

GAMERMAN, D., LOPES, H. F., 2006. *Markov chain Monte Carlo: stochastic simulation for Bayesian inference*. New York: Chapman and Hall/CRC .

HASTINGS, W. K., 1970. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1), 97–109.

HUSMEIER, D., 2005. Discriminating between rate heterogeneity and interspecific recombination in DNA sequence alignments with phylogenetic factorial hidden Markov models. *Bioinformatics*, 21(2), pp.ii166–ii172.

HUSMEIER, D., McGUIRE, G., 2003. Detecting recombination in 4-Taxa DNA sequence alignments with Bayesian hidden Markov models and Markov chain Monte Carlo. *Molecular Biology and Evolution.*, 20(3), pp.315–337.

LEGENDRE, P.,MAKARENKOV, V., 2002. Reconstruction of biogeographic and evolutionary networks using reticulograms. *Systematic Biology*, 51(2), pp.199–216.

MAKARENKOV, V., LEGENDRE, P., DESDEVISES, Y., 2004. Modelling phylogenetic relationships using reticulated networks. *Zoologica Scripta*, 33(1), pp.89–96.

MARTTINEN, P., BALDWIN, A., HANAGE, W. P., DOWSON, C., MAHENTHIRALINGAM, E., CORANDER, J., 2008. Bayesian modeling of recombination events in bacterial populations. *BMC Bioinformatics*, 9 (421).

METROPOLIS, N., ROSENBLUTH, A. W, TELLER, A. H., TELLER, E., 1953 Equation of state calculations by fast computing machines. *Journal of Chemical and Physics*, 21(6), pp.10871092.

PAGEL, M., MEADE, A., 2004. A phylogenetic mixture model for detecting pattern-heterogeneity in gene sequence or character-state data. *Systematic Biology*, 53(4), pp.571-581.

RIPLEY, B. D., 1987. *Stochastic Simulation*. New York: John Wiley and Sons.

RONQUIST, F., HUELSENBECK, J. P., 2003. MrBayes 3: Bayesian phylogenetic inference under mixed models. *Bioinformatics*, 19(12), pp.1572–1574.

STRIMMER, K., MOULTON, V., 2000. Likelihood analysis of phylogenetic networks using directed graphical models. *Molecular Biology and Evolution*, 17(6), pp.875–881.

TEMPLETON, A. R., CRANDALL, K. A., Sing, C. F., 1992. A cladistic analysis of phenotypic associations with haplotypes inferred from restriction endonuclease mapping and DNA sequence data III. Cladogram estimation. *Genetics*, 132(2), pp.619–633.